Development of a distributed Service Framework for Location-based Decision Support

by

Dipl. Ing. (FH) Johannes Sornig

Master Thesis

Submitted in partial fulfilment of the requirements of the degree
Master of Science in Engineering

Spatial Decision Support Systems: Geographic Information Science and Operations Research

At the Fachhochschule Kärnten
Carinthia University of Applied Sciences/School of Geoinformation

Supervisors:

Dr. Andreas Wytzisk
con.terra GmbH

Dr. Gernot Paulus
Carinthia University of Applied Sciences

Villach, 29th August 2008

Science Pledge

By my signature below, I certify that my thesis is entirely the result of my own work. I have cited all sources I have used in my thesis and I have always indicated their origin.

Villach, 29. August 2008

Signature

ABSTRACT

Progresses in the field of mobile Internet and usability of mobile devices have lead to new opportunities for location-based services. This thesis proposes the integration of multi-criteria decision analysis for mobile location aware applications. Location-based services equipped with decision support techniques provide users with a valuable extension in functionality and usability.

Humans make decisions day-to-day. Most of these decisions are related to a specific time and their current environment. People also tend to decide about problems when they occur and straight near the relevant location. In such cases mobile spatial decision support systems have significant advantages over traditional spatial decision support systems. Location-based decision services assist people in their decisions while they move through a physical environment. Such systems explicitly use decision support methods to suggest decision alternatives based on user preferences, the combination of multiple criteria and the current location.

For the research in mobile decision support theoretical aspects about decision analysis and actual trends in mobile computing and location-based services are evaluated and compared with each other. There is a momentous influence of the Internet and web-based technologies in both research areas. Hence, the most obvious design for a location-based decision service is open and distributed processing. This work introduces a comprehensive conceptual framework for implementing a location-based decision service. This framework is based on an n-tier software architecture, which is described from different viewpoints. Geographic information is a valuable resource for spatial decision making. Trends from the Web 2.0 fundamentally affect the geoinformation community. Due to user-generated content and volunteered geographic information we face an unheard-of ampleness of available and accessible information, useful also for decision making. The introduced prototype application combines user-generated content and expert data for evaluating suitable locations. The user has the possibility to rate different criteria on a mobile device to get a personal suggestion of decision alternatives. In this case the use of decision support enhances the personalisation of the location-based service. The application domain of the introduced prototype is tourism.

KEYWORDS:

Mobile Spatial Decision Support, Location Based Decision Service, Multi-criteria Decision Analysis, Tourist Guide Application, Spatial Decision Support System, Location Based Service.

KURZFASSUNG

Entwicklungen im Bereich des mobilen Internets und der Bedienbarkeit von mobilen Geräten führen zu neuen Möglichkeiten für standortbezogene Dienste. Diese Arbeit schlägt die Integration multi-kriterieller Analyse für mobile Standortdienste vor. Standortbezogende Dienste welche mit Entscheidungsunterstützungstechniken ausgestattet sind, bedeuten eine erhebliche Erweiterung der Funktionalität und Benutzbarkeit für den Anwender.

Menschen treffen laufend Entscheidungen. Die meisten dieser Entscheidungen stehen mit einer definierten Zeit und der aktuellen Umgebung im Zusammenhang. Der Mensch tendiert außerdem dazu, Entscheidungen unmittelbar beim Auftreten und in räumlicher Nähe des Problems zu treffen. In solchen Fällen haben mobile räumliche Entscheidungsunterstützungssysteme einen entscheidenden Vorteil gegenüber traditionellen Entscheidungsunterstützungssystemen. Standortbezogende Entscheidungsdienste unterstützen Menschen in ihren Entscheidungen während sie sich durch eine physikalische Umgebung bewegen. Solche Systeme nutzen explizit Entscheidungsunterstützungsmethoden um Alternativen, welche auf Vorzüge des Benutzers, eine Kombination aus mehreren Kriterien und die aktuelle Position eingehen, vorzuschlagen.

Für die Forschung im Bereich der mobilen Entscheidungsunterstützung wurden theoretische Aspekte über Entscheidungsanalyse und Trends über standortbezogene Dienste ausgewertet und miteinander verglichen. Das Internet und web-basierende Technologien üben hier einen beträchtlichen Einfluss auf beide Themengebiete aus. Daher ergab sich als offensichtliche Designvariante für die Entwicklung eines standortbezogenen Entscheidungsdienstes ein offenes und verteiltes Verarbeiten der Teilaufgaben des Systems. Die Arbeit schließt ein konzeptionelles Rahmenwerk für die Implementierung standortbezogender Dienste ein. Dieses Rahmenwerk basiert auf einer Mehrschichtarchitektur die aus verschiedenen Blickwinkeln beschrieben wird. Geographische Information ist eine bedeutende Ressource zur räumlichen Entscheidungsfindung. Entwicklungen betreffend *Web 2.0* beeinflussen die Geoinformationsgesellschaft bedeutend. Wegen benutzergenerieten Inhalten und freiwillig erstellter geographischer Information stehen wir einer nie dagewesener Fülle an verfügbarer und zugänglicher Information gegenüber. Dieses Potential ist durchaus für Entscheidungsfindungen nützlich. Die vorgestellte Prototyp Anwendung kombiniert benutzergenerierten Inhalt mit Expertendaten um passende Standorte zu bewerten. Der Benutzer des Systems hat die Möglichkeit verschiedene Kriterien am mobilen Endgerät zu beurteilen um personalisierte Vorschläge zu Entscheidungsalternativen zu erhalten. In diesem Fall erhöht die Integration von entscheidungsunterstützenden Methoden die Personalisierung von standortbezogenen Diensten. Als Anwendungsgebiet für den Prototyp wurde Tourismus gewählt.

ACKNOWLEDGEMENTS

I want to thank my advisor Dr. Andreas Wytzisk for his supervision, and Dr. Gernot Paulus for co-supervising my thesis. Their valuable comments and professional notes assist me throughout the whole work. Furthermore, I appreciate the collaboration with the entire team working for the *School of Geoinformation* at the *Carinthia University of Applied Sciences*. I thank Michael Diener and Stefanie Andrae for their encouragement to work in the mobile GIS sector and their discussions and input about different actual aspects about Geoinformation.

I am gratefully to many friends and colleagues who accompanied me throughout the study and my thesis. With our (sometimes funny) discussions over a beer or coffee Patricia Steiner and Christoph Erlacher motivated me during annoying phases of the study.

Additionally, I want to express my deep appreciation for the vast and creative geospatial community. Their ideas, work and communication of geospatial knowledge contribute to the great popularity of “Geo” in information technology and make Geoinformation such an interesting field to work.

And most importantly, I admire the support of my family. My parents gave me the chance to go my own way and keep me always grounded. Without their support it would not be possible to work on this thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

.NET CF	Microsoft's .NET Compact Framework
3G	third Generation of mobile phone standards and technology
ADT	Abstract Data Type
AHP	Analytic Hierarchy Process
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
ASP	Active Server Pages
BIM	Building Information Model
CGI	Cell Global Identify
CORBA	Common Object Request Broker Architecture
DBMS	Database Management System
DCOM	Distributed Common Object Model
DoD	U.S. Department of Defense
DoT	U.S. Department of Transportation
DSS	Decision Support System
ESRI	Environmental Systems Research Institute
GeoDRM	Geospatial Digital Rights Management
GIS	Geographic Information System
GLONAS	Global Orbiting Navigation Satellite System
GML	Geography Markup Language
GMS	OGC GeoMobility Server
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSM	Global System for Mobile Communication
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
IDE	Integrated Development Environment
IEC	International Electrotechnical Commission
IIS	Internet Information Server
ISO	International Organization for Standardization

ITU-T	Telecommunication Standardization Sector of the International Telecommunication Union
JSON	JavaScript Object Notation
JSP	Java Server Pages
KML	Keyhole Markup Language
LAN	Local Area Network
LBDS	Location Based Decision Service
LBS	Location Based Service
MADM	Multi-Attributive Decision Making
MCDA	Multi-Criteria Decision Analysis
MCDM	Multiple Criteria Decision Making
MCE	Multi-Criteria Evaluation
MC-SDSS	Multi-Criteria Spatial Decision Support System
MODM	Multi-Objective Decision Making
MPS	Mobile Positioning System
mSDSS	mobile Spatial Decision Support System
NASA	National Aeronautics and Space Administration
NICT	New Information Communication Technology
ODIS	Ontology-Driven Information System
ODP	Open Distributed Processing
OGC	Open Geospatial Consortium
OMA	Open Mobile Alliance
OLAP	Online Analytical Processing
OWA	Ordered Weighted Averaging
PDA	Personal Digital Assistant
PDE	Position Determination Equipment
PHP	Hypertext Preprocessor
POI	Point of Interest
REST	Representational State Transfer
RMI	Remote Method Invocation
RM-ODP	Reference Model of Open Distributed Processing
RPC	Remote Procedure Call
RSS	Really Simple Syndication
SAW	Simple Additive Weighting
SDI	Spatial Data Infrastructure
SDK	Software Development Kit
SDSS	Spatial Decision Support System
SFS	Simple Feature Specification
SMS	Short Messaging Service

SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SVG	Scalable Vector Graphics
SWE	Sensor Web Enablement
UDDI	Universal Description, Discovery and Integration
UI	User Interface
UML	Unified Modeling Language
UMTS	Universal Mobile Telecommunications System
VGI	Volunteered Geographic Information
W3C	World Wide Web Consortium
WGS84	World Geodetic System of 1984
WLAN	Wireless Local Area Network
WLC	Weighted Linear Combination
WMS	Web Map Service
WSDL	Web Services Description Language
XML	eXtensible Markup Language
XML-RPC	eXtensible Markup Language Remote Procedure Call

1. INTRODUCTION

"Knowing where things are, and why, is essential to rational decision making."

Jack Dangermond

Humans make decisions throughout the whole day, aware or not aware, ranging from trivial to complex or related to space and time. Decision making can be regarded as an outcome of cognitive processes leading to the selection of several alternatives and act or have an opinion on one of these alternatives. Every decision making process produces a final choice (Reason, 1990). Hence, decision making is a reasoning or emotional process which can be rational or irrational. Logical decision making is an important part of different sciences including the field of decision support, with the goal to structure and reconstruct the decision making process in order to create supporting systems.

The introduction of this thesis will dwell on the motivation for designing and using a mobile Spatial Decision Support System (mSDSS). This section covers the outline of the project including hypotheses, the scope and objectives. Finally the organization and structure of the thesis is described.

1.1. MOTIVATION

Geographic information is a valuable resource for decision making processes in different areas ranging from urban planning, emergency response, and environmental analysis to name a few. For this reason spatial decision support is an important subset in decision support sciences. The combination between Geographic Information Systems (GISs) and Decision Support Systems (DSSs) is suitable to support spatial decision problems. Spatial Decision Support Systems (SDSSs) have emerged from the co-evolution of research in DSSs and GISs. Because of actual issues about interoperability and scale, these systems face new challenges and opportunities in an interconnected world (Casey and Austin, 2002). Rapidly developing interactive computer technology, especially web technology, allows improving the usefulness of decision models. The web has the potential to deliver GIS and decision support technologies to the masses. Advances in communication and networking technologies (Internet, Wireless and Cellular) will facilitate the development and deployment of inter-organizational SDSSs that support spatial work-flows. In recent years the amount of available and accessible geospatial data, which can be used for decision support, has increased significantly. However, a major drawback for DSSs in general is the communication between the system and the decision maker. Decision makers must articulate well their information needs, and the models must communicate effectively their results. This

communication is increasingly possible now that interactive, user-friendly computer systems have become the rule, rather than the exception.

There is a big trend to combine DSSs which incorporate simulation and optimisation models with interactive graphics capabilities to encourage the acceptance of these techniques in practice. GIS traditionally has very strong capabilities in interactive graphical visualisation and simulation and is well suited for a combination with DSSs. For desktop computing and more recently standard web services SDSSs are used to assist decision makers in solving complex and large scale decision problems. These systems could also assist the decision maker in the evaluation of reliability of the results and the generation of alternatives.

Unlike desktop computing and the *large screen* web the development of decision support methods for mobile platforms lag far behind in case of sophistication. There is an extensive fragmentation on the mobile handset market – with devices varying widely in terms of screen size and processing power, as well as in the programming and mark-up languages they support – that makes the creation of cross-device-compatible mobile applications a tedious and costly task. Position technologies are inconsistently supported and implemented, tools are complex and standardisation is still in a state of evolution (Simon, 2008). Beside these drawbacks it would be extremely useful to integrate spatial decision analysis in mobile geospatial or location aware applications. Many spatially related problems occur while moving through a physical environment, and people tend to decide on problems when they become relevant. A mobile device providing a supporting application can assist people in finding a decision, related to their actual situation, near real-time.

Location Based Services (LBSs) are spatial information services accessible via a mobile device. In many cases the goal of LBSs is to provide information for decision support, but LBSs generally do not include explicit decision support methods. With the introduction of the Universal Mobile Telecommunications System (UMTS) a big hype for LBSs was predicted, but the potential of LBSs is still not reached. The trend of LBS applications shows that stronger personalization of the services is necessary to provide adequate and user specific results (Zipf and Strobl, 2002). At the same time the privacy of the user has to be ensured. LBS applications have found insufficient in considering individual user preferences and possible subtasks (Zipf and Strobl, 2002, Rinner and Raubal, 2005). To add decision support methods is one way to integrate personalization in such services. LBSs should not only inform the users about spatial phenomena but also assist them in their decision processes according a task related to space and eventually to time. The definition of Rinner and Raubal (2005) about LBSs focuses on this idea: *"A LBS assist people in decision-making while they perform tasks in space and time"*.

Decision support methods in GISs and LBSs go beyond simple querying. They enable users to evaluate and rank decision alternatives based on multiple criteria. GIS-based Multi-Criteria Evaluation (MCE) is commonly used in applications such as site suitability analysis (Malczewski, 1999). This set of methods, leads to new challenges to software applications for mobile devices, and shows that LBSs have to be enhanced in case of decision support functionality and representation of decision alternatives. Research in geographic information science and related disciplines like SDSSs or information technology can be considered for designing LBSs that assist people in their decision processes. LBSs with decision support methods are sometimes named as Location Based Decision Services (Pühretmair *et al.*, 2002, Raubal, 2006, Muntermann, 2007) or mobile Spatial Decision Support Systems (mSDSS). There are attempts to integrate spatial decision support methods in web services (Rinner and Malczewski, 2002, Rinner, 2003, Sugumaran and Sugumaran, 2005), but until now decision support methods in combination with

LBSs are rarely suggested (Lee, 2005, Rinner and Raubal, 2005). Therefore, in this work an exemplary framework is presented to integrate decision support methodology into LBSs.

1.2. PROJECT DESCRIPTION, SCOPE AND GOAL

This work describes user-guided methods and user interaction to maps on mobile devices to answer spatial and location related questions and support decision processes. Decision analysis is a set of systematic procedures for analyzing complex decision problems. The basic strategy is to divide the decision problem into small understandable parts; analyze each part; and integrate the parts in a logical manner to produce a meaningful solution (Malczewski, 1999). The integration of web services and user generated content can simplify the decision process and reduce the complexity of the overall decision analysis. Especially information, produced by a community and accessed via a web service, can be useful to integrate in the decision analysis process. For example, a service where the community rate different hotels can be integrated in a LBS application for finding the best hotel according to different user parameters. The integration and combination of distributed web services for data and functionality is influencing the represented result. In this approach user-generated information together with actual location information is used as parameter influencing the result of an mSDSS. Additional parameters for LBSs are evaluated in order to enhance the user personalization. Location Based Decision Services (LBDSs) include decision alternatives which are considered and integrated in the system. This work shows how web services can be part of an LBDS and simplify the decision process. Decision strategies are reconsidered in order to integrate them into mobile applications.

Users should be supported in trading off good against poor characteristics of alternative destinations for example in a multi-criteria evaluation or other decision support methods. MCE is a decision support methodology, which is based on the idea that people use multiple decision criteria to determine their best solution. Multi-criteria decision rules have been implemented in GISs since the 1990s including the simple additive weighting, analytic hierarchy process, ideal point analysis, concordance, and ordered weighted averaging methods (Carver, 1991, Jankowski, 1995, Malczewski, 1999, Jankowski and Nyerges, 2001). An investigation on existing LBSs and academic research is done to evaluate the state of the art. Different decision support methods e.g., multi-criteria decision making, will be reviewed for integration in mobile geographic applications. The differences between current LBS approaches and LBSs with decision support and the motivation to integrate spatial decision support techniques is shown.

Recent developments in the sector of mobile devices and mobile phones, which can be seen as secondary or enabling technologies for spatial and location services, should be considered to build new user centralized, mobile decision support systems. These secondary or enabling technologies are for example touch- or multi-touch screens and gesture interaction, open frameworks for mobile platforms; (relatively) fast internet connections (3G and Wi-Fi) and positioning technologies. The work should show a possible architecture for building LBDSs or mSDSSs.

Objective of this work is to show the motivation and advantages for integrating decision support methods in LBSs. In this scope current LBS architectures are evaluated and differences between current approaches and LBDSs or mSDSSs are identified. *Geoweb* services or *geoweb 2.0* services can simplify the decision analysis. As output a proposal is given how these services can be integrated in LBS applications to reduce the complexity of the decision strategy and serve for

decision support. One part is to show the combination and interaction between user generated information and decision support methods for LBSs.

The design of a prototype will show the possibilities of distributed web services on a mobile map interface of an LBDS. The example allows input for multi criteria evaluation. Users specify decision relevant parameters to be used as evaluation criteria. The current location of the device is also used as significant parameter in the decision process. This prototype is evaluated and compared with existing solutions showing advantages and disadvantages. The solution should be open and flexible to extent features and modify it for different application fields, and should serve users with information that support their decisions near real time.

1.3. HYPOTHESES

This thesis contributes to the foundation of decision support and location-based services by addressing research issues to integrate decision support functionality into location-based services. State of the art applications are investigated to identify future trends in the covered areas. One major trend is the personalisation of mobile location aware applications, the first hypothesis therefore is:

> *Personalisation of mobile geoinformation services can be done with the integration of spatial decision support methods.*

Secondly, this work discusses the influence of web technologies in the field of decision analysis. New paradigms like the *Web 2.0* with user-generated content can influence decision support. Of special interest are user-generated geographical content and Volunteered Geographic Information (VGI), which plays an increasingly important role in Spatial Data Infrastructures (SDIs) in the coming years and their role for web-based SDSSs. The second hypothesis is derived from these considerations:

> *Geoweb services and geoweb 2.0 services can simplify the decision process and reduce the complexity of the overall decision strategy in a location-based decision service.*

These two hypotheses are discussed throughout the whole work to verify the statements.

1.4. INTENDED AUDIENCE

This thesis is intended for the geoinformation and decision analysis community, especially for people interested in mobile services and location based services. It is targeted to people with GIS background who are interested in decision support techniques, both from the theoretical and practical side. The practitioners will find solutions and indications for many questions that may arise when handling such applications, from a high-level viewpoint and from a technical viewpoint. Researchers will discover the breadth and depth of numerous research challenges in the different areas concerned.

Some information about spatial decision support in general and multi-criteria decision methods to decision support are given, but it is not intended to give a complete overview about spatial decision analysis. Additionally the work focuses on location based services and the integration of server side processing routines via network communication. The intention of this thesis is to give the reader an overview about the use of multi-criteria decision techniques within a mobile location aware application.

1.5. STRUCTURE OF THE THESIS

The remainder of the thesis is organized into seven major sections covering the theoretical and practical parts of the work:

SECTION 2: LITERATURE REVIEW AND SCOPE

The following chapter provides an introduction to location-based services and spatial decision support systems. It gives an overview about current technologies and design strategies for LBSs and SDSSs. One subsection focuses on decision analysis in general. The section *Literature Review and Scope* describes the technological and scientific background for the work presented in this thesis.

SECTION 3: METHODOLOGY

This chapter emphasised on background about spatial decision support for location-based services. The chapter *Methodology* includes a definition for location based decision services and gives examples for applications. Technologies, which may influence the design of location-based decision services and the architecture of such systems is described.

SECTION 4: CONCEPT AND SYSTEM DESIGN

The section *Concept and System Design* focuses on the idea to combine LBSs and SDSSs for mobile-based decision support. Design ideas from both sides are brought together to model a distributed and service oriented mobile decision support system.

SECTION 5: PROOF OF CONCEPT

In this chapter the presented concepts are realized in a LBS prototype application using MCE. Concrete technologies and programming languages for the different parts of the applications are described.

SECTION 6: RESULTS

The *Results* describe conclusions from the architecture and framework to combine spatial decision support methods with LBSs to implement a LBDS system. Additional the functionality of the prototype application is described as LBDS which includes MCE to display a personal list of favourite places.

Section 7: Conclusion and Further Directions

This chapter summarize the whole thesis and gives an outlook to further directions and not covered considerations.

Section 8: Literature

This chapter includes a complete list of bibliographical references used in this thesis.

Proper names and words in the context having a special meaning are written *italic*. Mathematical expressions and formulas are written with font type Cambria Math and source code is represented in `Courier New`. A literal citation is marked with *"quotations and is written in italic"*.

2. LITERATURE REVIEW AND SCOPE

"Research is to see what everybody else has seen, and to think what nobody else has thought."

Albert Szent-Gyoergi

This chapter gives a literature review about the topics of Location Based Services (LBSs) and Spatial Decision Support Systems (SDSSs). Recently these fields are used for many different applications, due to the fact of rapidly change in technology both hardware and software. Also the Internet has a huge influence to both of these disciplines. In the following part theory behind LBS and SDSSs as well as functionality and architecture are described.

2.1. LOCATION BASED SERVICES

Location services can be defined as services that integrate the location or position of a mobile device with other information so as to provide added value to the user. Schiller and Voisard (2004) describe the term LBS as a recent concept that denotes applications integrating geographic location with the general notion of services. LBSs can be seen as location aware services or applications with the general ability to infer information about the physical location of its user and to adapt its information offering or behaviour in response to this information.

2.1.1. INTRODUCTION TO LBS

Location services have a long tradition, related to the Global Positioning System (GPS), which has been operating since the 1970s by the U.S. Department of Defense. GPS[1] is a Global Navigation Satellite System (GNSS) or satellite infrastructure, which allows determining the position on the earth surface via a device, connected with satellites. However, widespread interest in LBSs started in the late 1990s, when mobile networks where widely deployed in Europe, North America and Asia. One of the first research efforts dealing with location and handheld devices was the work on the ParcTab system (Want *et al.*, 1995). The wide range of applications including car navigation systems, tourist tour planning and tourist information systems, yellow pages and maps,

[1] Beside the GPS system, the former USSR has started a comparable system called Global Orbiting Navigation Satellite System (GLONASS), and the European Union is also planning an own system called GALILEO. For more information see chapter 2.1.3.

emergency services and disaster management or buddy finder and instance messaging. A range of different application fields for LBS applications is listed in Figure 2.1.

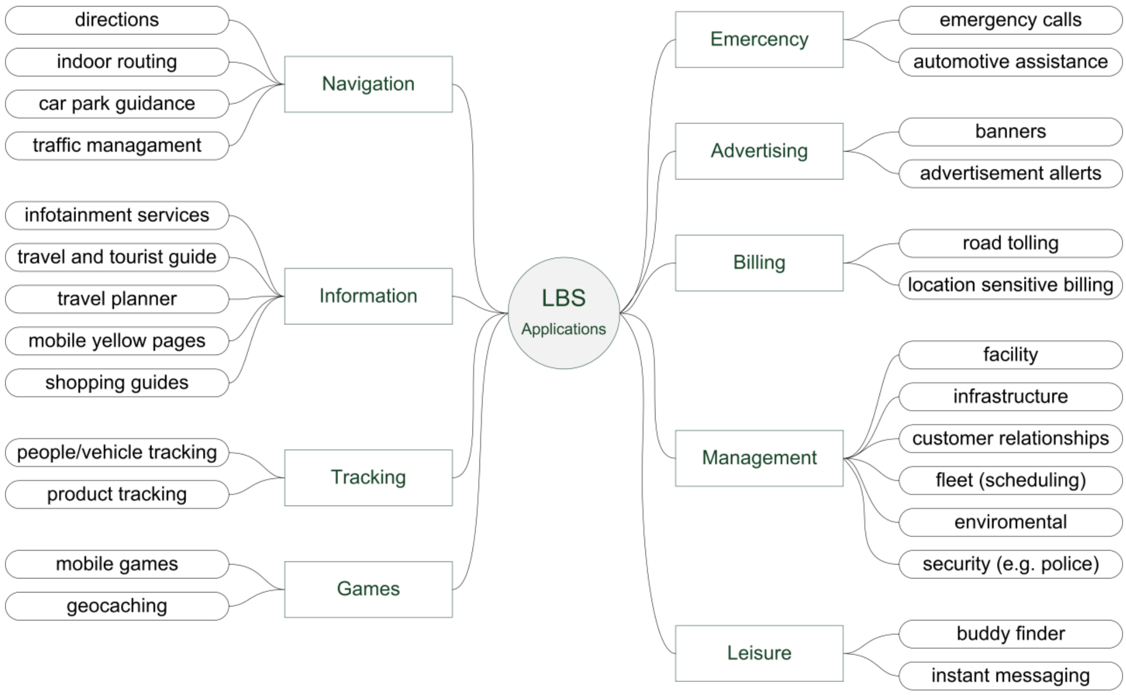

Figure 2.1: LBS applications.

Modern LBSs are defined as information services accessible with mobile devices through a mobile network and utilizing the ability to make use of the location of the mobile device (Virrantaus *et al.*, 2001). While other definitions describe LBSs as wireless service that uses geographic information to serve a mobile user. The *Global System for Mobile Communication* (GSM) family was the first mobile phone standard allowing wireless communications for an enormous number of people in many parts of the world. A mobile or wireless communication network is an important factor for all LBS applications, and successor technologies of GSM like the *Universal Mobile Telecommunications System* (UMTS) are even better suited for LBSs. A LBS is an intersection between internet technologies, new information communication technologies and GIS/spatial database technologies (Figure 2.2). Research topics related to LBSs include network architectures and standards (Adams *et al.*, 2003, Ahn *et al.*, 2004, Beaubrun *et al.*, 2007), mobile positioning techniques and space-time recording (Mountain and Raper, 2001, Miller, 2003, Worboys and Duckham, 2004), user interface design and user interaction (Meng, 2001, Hjelm, 2002, Zipf and Strobl, 2002), business cases and application fields for LBS (Beinat, 2001, Benson, 2001, Barnes, 2003) and locational privacy (Kwan and Schmitz, 2002, Myles *et al.*, 2003, Armstrong and Ruggles, 2005).

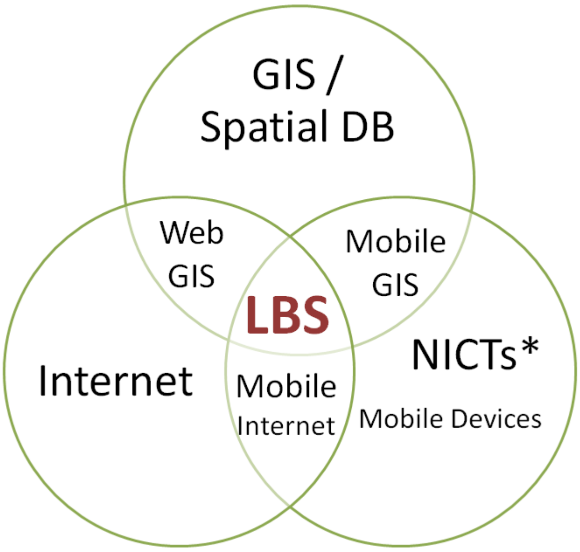

Figure 2.2: LBS and related technologies (Brimicombe, 2002).

After the unexpected success of the Short Messaging Service (SMS) for GSM, LBSs have been promoted as the next *killer* application since the introduction of the 3rd generation mobile communication systems in the late 1990s (Simon, 2008). Despite these expectations until today the predicted success is absence, including wide-spread user acceptance and commercial success (Uhlirz, 2007). Nevertheless current user studies and market forecasts are still positive and predict a high demand for LBSs. Some studies[2] expect total global LBS revenue of 13.3 billion USD in 2013. This shows a remarkable increase in revenue when it is suggested that during 2007 a total global revenue of 515 million USD is estimated.

Mobile geospatial applications like maps or driving directions are marketed by high-end mobile phones, with included GPS receiver. Due to the decreasing costs for GPS receiver chips and alternative positioning techniques geospatial applications will be integrated in most mobile phones. Marcussen (2000) describes ten factors which are necessary to ensure the success of LBSs. These could be summarized in bandwidth, actual content, location, appropriate costs and secure payment, usability, personalization and privacy, portals and search engines and internet enabled mobile devices. This list shows aspects that have to be considered for implementing LBSs.

2.1.2. LBS Application Taxonomy

According to the viewpoint there are several possibilities to distinguish between different kinds of LBSs. A major distinction of services is weather a service is person oriented or device oriented (Schiller and Voisard, 2004):

[2] Mobile Location-Bases Services: Market Development, Revenue Opportunities, LBS Applications, and Key Industry Players, Report from Allied Business Intelligence, Inc., 2008.

- *Person oriented* LBSs compromise all of those applications where a service is user-based. This kind of services is used to position a person or the position of a person is used to enhance the service (e.g., friend finder application).
- *Device oriented* LBS applications are external to the user. Instead of a person also an object or group of people could be located. In device oriented applications the person or object located is usually not controlling the service (e.g., car tracking for theft recovery).

One further classification is done on the application design (Schiller and Voisard, 2004, Kolodziej and Hjelm, 2006). Here in general two different kinds of location services, considering if information is delivered on user interaction or not, can be distinguished:

- *Push services* imply that the user receives information without direct or active request. The information may be sent to the user with prior consent (e.g., subscription-based) or without prior consent (e.g., advertising message)
- *Pull services* deliver information actively requested from the users. That means a user actively uses an application and pulls information from the network. For pull services a further separation can be done into functional services, like ordering a taxi by using a service on the device, or information services, like the search for Point of Interests (POIs).

Some services such as a friend finder application integrate both push and pull functionality. Table 2.1 gives some practical examples for push and pull services, either person oriented or device oriented.

Table 2.1: Categories and examples of LBS applications (Schiller and Voisard, 2004).

	Push Services	**Pull Services**
Person oriented		
Communication	Ex. 1: You get an alert from a friend zone application that a friend has just entered your area. Ex. 2: A message is pushed to you asking whether you allow a friend to locate you.	Ex. 1: You request from a friend finder application persons who are near you.
Information	Ex. 3: You get an alert that a terror alarm has been issued by the city you are in.	Ex. 2: You look for the nearest cinema in your area and navigation instructions to get there.
Entertainment	Ex. 4: You have opted to participate in a location-based "shoot 'em up" game and are been attacked.	Ex. 3: You play a location-based game and look for another opt-in in your area to attack.
M-Commerce and Advertising	Ex. 5: A discount voucher is being sent to you from a restaurant in the area you are in.	Ex. 4: You look for cool events happening in the area you are in.
Device oriented		
Tracking	Ex. 6: An alert is sent to you from an asset-tracking application that one of your shipments has just deviated from its foreseen route. Ex. 7: You get an alert that your child has left the playground.	Ex. 5: You request information on where your truck fleet currently is located in the country.

Most LBSs include spatial queries, others use a combination between spatial queries and attributive information.

2.1.3. POSITIONING TECHNIQUES

LBSs must be able to detect the geographic location of a mobile user. Positioning and navigation have a long history. As long as people move, they want to determine their actual location. Nowadays there are several possibilities available to find out the current position, but all these techniques have advantages and disadvantages. Until now there is no positioning system available that fulfils all of the needs of any LBS. A satellite based positioning system achieves high coverage and precision, but it fails in indoor environments. Indoor positioning techniques require cost intensive installations and are restricted to buildings and are restricted to a limited area of coverage. In this chapter the large area of positioning system is limited to techniques that are applicable for LBSs.

Schiller and Voisard (2004) summarizes the properties of location data with following characteristics:

- *Coordinate system*. Coordinate systems that describe a 3D worldwide unique location are necessary for most LBSs. They can be divided into latitude, longitude and altitude systems and earth cantered, earth fixed systems that use Cartesian coordinates.
- *Scope*. A positioning system has a certain scope and defines an area of potential coordinates. A location can be worldwide unique or only valid in a small area.
- *Coverage*. The actual coverage of a location system may be smaller than the area of potential locations specified by the scope.
- *Precision*. A positioning system produces certain measurement errors while capturing a location. Users and services that access that location information must be aware of inaccuracy.
- *Geographic vs. semantic location*. Users of LBSs are often interested in the meaning of a location rather than the geographic coordinates.
- *Additional spatial data*. Beside location additional spatial data may be of interest, e.g., orientation, speed, etc.

Following subsections are based on Schiller and Voisard (2004) and give a short overview about satellite positioning systems, indoor positioning systems and systems that used existing network infrastructure. Therefore references are not explicitly declared in this section.

SATELLITE POSITIONING SYSTEMS

The idea of using satellites for positioning tasks arises in the 1960s. Figure 2.3 illustrates the basic principle of satellite positioning. A person who wants to determine a position with a GNSS needs the exact positions of the satellites (s_i) as well as the exact distances to the satellites (r_i). The position of the user is restricted to the spherical surfaces around each satellite. To determine the location in three dimensions at least three satellites are necessary. The cut of three spherical surfaces normally leads to two intersection points, but one of these points lies far in the space. Therefore it can be simply referred to the right intersection point near the earth's surface.

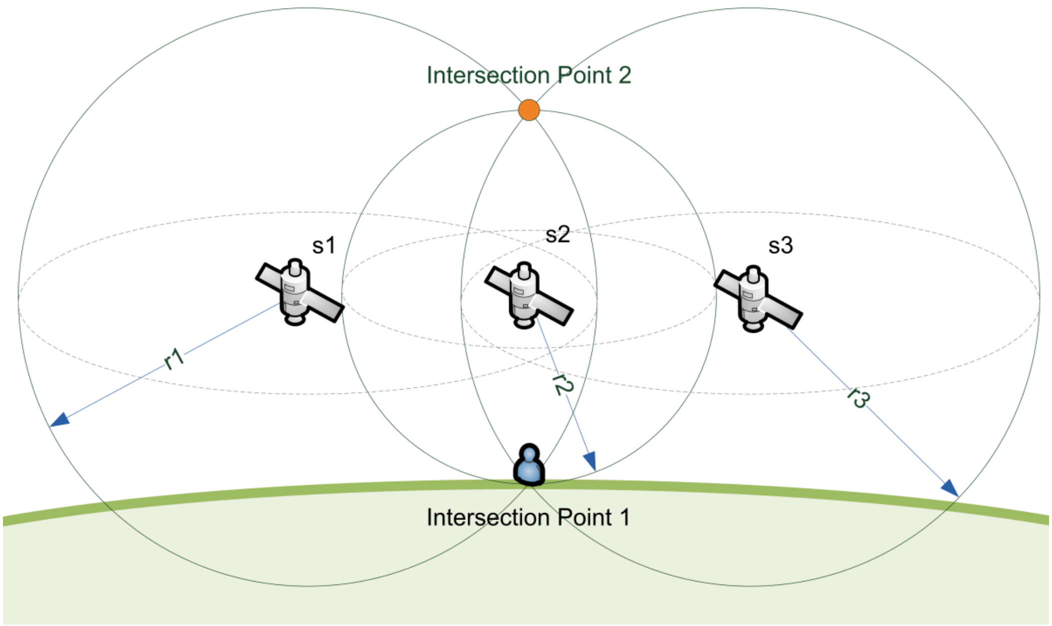

Figure 2.3: Principle of satellite positioning.

The most prominent example of a satellite navigation system is GPS. Different American organisation, among them the Department of Defense (DoD), the Department of Transportation (DoT), and the National Aeronautics and Space Administration (NASA), where interested in a satellite based positioning system in the early 1960s. In 1984 the first satellites where launched and in 1995 the full operation capability was declared. The GPS is divided into three segments:

- The *user segment* contains a GPS receiver, which can be build-in chips or separate devices with a serial interface connection.
- The *space segment* consists of the satellites, which has a central computer and an autonomous energy supply with solar cells.
- The *control segment* is necessary for the administration of the satellites as well as for correction of the satellite internal data like system time and orbits.

Beside the GPS system, the former USSR has started a counterpart to GPS called Global Orbiting Navigation Satellite System (GLONASS), and the European Union is also planning an own system called GALILEO.

INDOOR POSITIONING SYSTEMS

Satellite navigation unfortunately, can only used outside of buildings because the radio signals employed cannot penetrate solid walls. This is the basic motivation to install indoor positioning systems. Techniques for indoor positioning systems are very different concerning mechanisms, precision or costs. They can be classified by measuring techniques based on infrared, radio, ultrasound and video:

- *Infrared Beacons*. The user carries an infrared device which pulse signals all the time. This signal transports a code that specifies the user's identity. Infrared sensors installed inside the building receive the signal and pass the information to a computer in the building. The mechanism is based on the fact that infrared signals normally are limited to a single room. Inside a room, walls reflect the infrared signal, thus a sensor can even receive a signal if it

has no direct signal to the user. The principle of an indoor position system based on infrared beacons is shown in Figure 2.4.

- *Radio Beacons*. Radio signals, in contrast to infrared signals, can penetrate walls. If more than one radio transmitter is in reach, a physical location can be calculated from the signals. In principle, positioning is similar to satellite navigation with several fixed radio beacons instead of satellites.
- *Ultrasound systems*. Systems based on ultrasound can achieve very high precision because the time an ultrasound signal needs from transmitter to receiver is approximately proportional to the corresponding distance. A user carries a device, which sends a short ultrasound impulse upon a request to the server. The server transmits the send request by radio. Receivers are attached in a raster. When they receive a signal, they immediately pass this information to the location via a server. The server has all of the necessary information about the positioning of the corresponding user.
- *Video Based Systems*. Another class of positioning systems is based on the evaluation of video data. Detecting the user's location can be done in two ways. First, the building can be equipped with cameras, which look for visual tags in their data stream. If at least two cameras detect the same tag, they can determine the corresponding angles under which the tag is visible. Because the positions of the cameras are known, they can compute the user's location by triangulation. Another approach is where the user itself is equipped with a camera. In this case visual tags are attached on walls inside the building and have fixed positions. If the camera detects two or more tags, it can find out its own position.

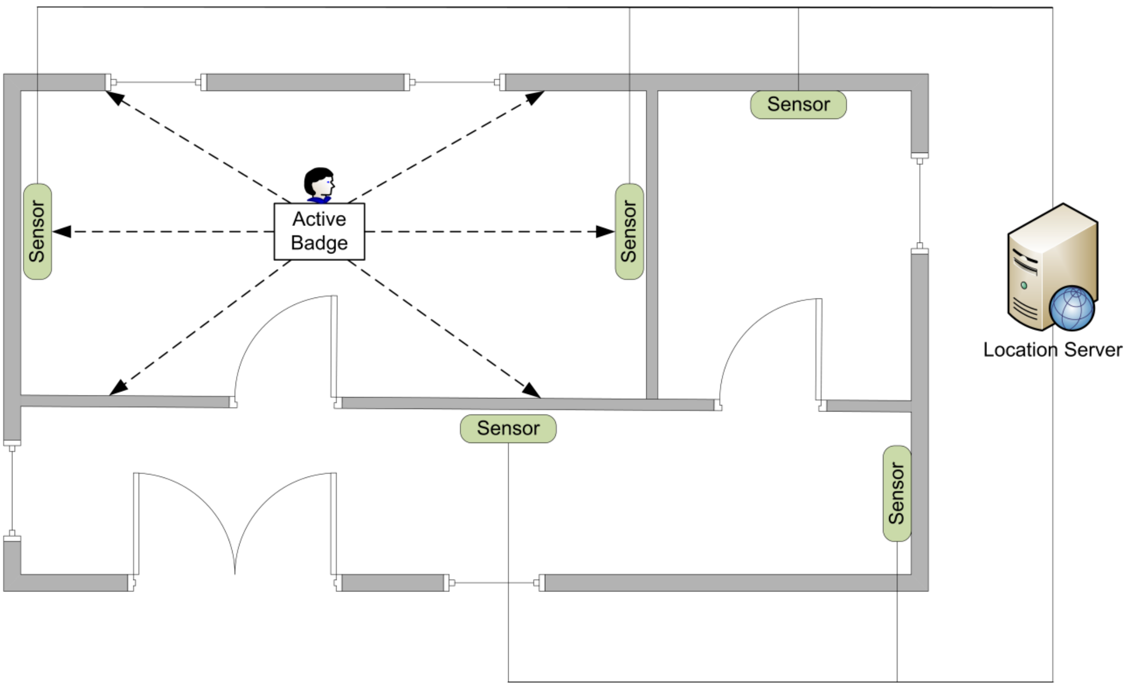

Figure 2.4: Infrared beacons as indoor position system.

NETWORK-BASED POSITIONING SYSTEMS

To reduce the investments of installing a new positioning infrastructure existing wireless networks are used for positioning services. Particularly cellular networks are suitable for positioning purposes because the cell identification already transports a rough location.

Additional mechanisms such as runtime measurement or angle measurement enhance the precision of the location.

A popular standard for cellular phone service used for positioning is GSM, because it is highly available, covers a large geographic area and reaches a high number of users. Without any further installations, a simple positioning is possible within the GSM network, which knows exactly in which cell which phone is registered. The resolution of the position coming from this method might be too inaccurate for some services, because the cell radius varies from less than 1 kilometre to more than 35 kilometres. Several techniques where developed to make more exact positioning possible e.g., MPS[3]. Different types of MPS are shown in Figure 2.5.

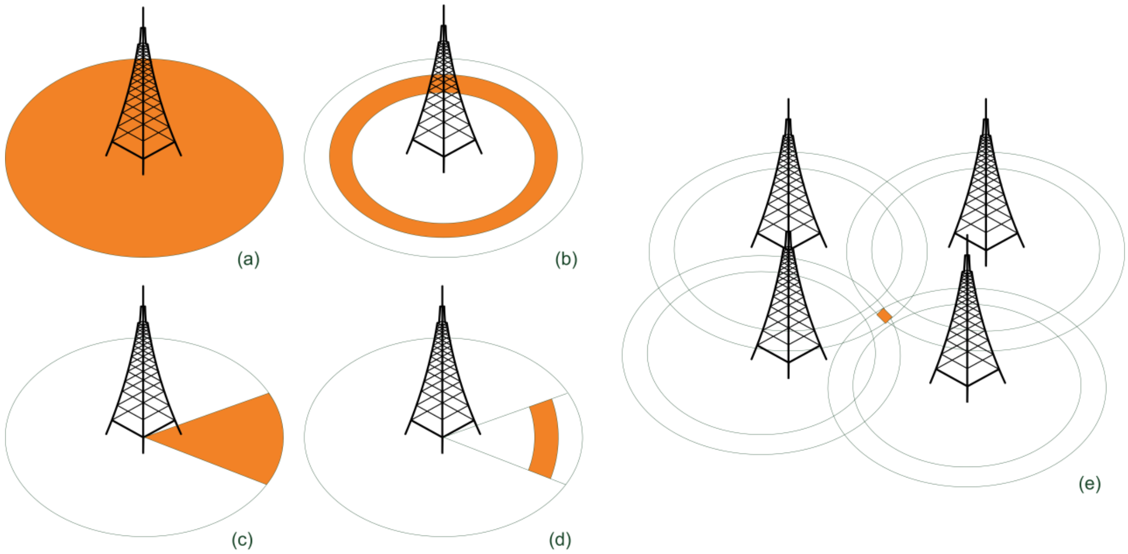

Figure 2.5: Positioning with GSM networks. (a) Cell Global Identify (CGI) mechanism. (b) CGI in combination with Timing Advance. (c) CGI segment antennas. (d) CGI and segment antennas and Timing Advance. (e) Uplink Time of Arrival method.

Another type of positioning system uses an available wireless LAN infrastructure. Inside the network of a WLAN infrastructure, where the access points are geo-referenced, the location of the user can be find out by measuring the signal strengths of all access points. Currently companies built database with locations of public WLAN hotspots to provide a positioning service. This kind of positioning techniques works fine in urban regions with a high density of access points and could be an addition to GPS which works fine far away from civilisation.

To summarize different positioning systems Table 2.2 categorizes and compares different mechanisms and gives an overview about their precisions.

[3] Mobile Positioning System (MPS) was developed by Ericsson and makes it possible to retrieve more accurate positions from large cells. MPS operates with standard GSM and needs some minimal modifications for installation at the communication infrastructure.

Table 2.2: Comparison of positioning systems.

Name	Category	Medium	Precision
GPS	Satellite	Radio	25 m
DGPS	Satellite	Radio	3 m
Active Badge	Indoor	Infrared	Cell
Active Bat	Indoor	Ultrasound/Radio	0.1 m
Visual Tags	Indoor	Video	Depends on camera resolution
GSM	Network	Radio	Cell
MPS	Network	Radio	150 m
Nibble	Network	Radio	3 m

Most LBSs use GPS to determine the current location but there are some alternative positioning techniques available, which have advantages for special applications. To reduce disadvantages of a single system different techniques are sometimes combined to reach the objective of a LBS application.

2.2. COMMUNICATION AND DATA MANAGEMENT IN LBS

In the past several years technical challenges have to be faced in order to design and implement a LBS. Technologically, a LBS can be described as a three-tier communication model, including a positioning layer, a middleware layer, and an application layer (Figure 2.6).

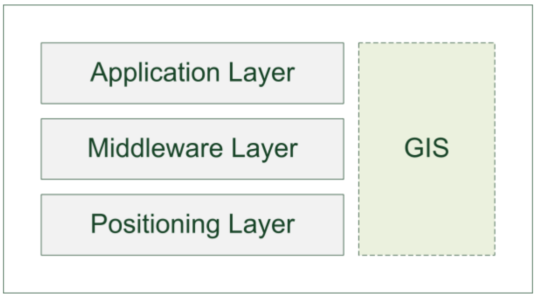

Figure 2.6: General LBS communication model (Schiller and Voisard, 2004).

The *positioning layer* is responsible for the determination of the geographic position of a mobile device or user. While a Position Determination Equipment (PDE) calculates the location in network terms a GIS core allows translating the raw information in geographic information (e.g., longitude and latitude). This result is then passed via a location gateway directly to an application or middleware layer.

For many LBS applications the network operators have put a *middleware layer* between the positioning layer and application layer. Request for location of the PDE is complex and lengthy and the integration of a middleware layer can reduce the complexity of service integration because it is connected with the operator network and can control location services. Location middleware can be used to manage interoperability between networks for location data.

The *application layer* comprises all of those services that request location data to integrate it into their offering. On this layer the actual software and user interface is implemented.

2.2.1. SPATIAL DATA REPRESENTATION FOR LBS

A general trend is seen to represent and visualize spatial information via *eXtensible Markup Language* (XML) structures. Many different disciplines, including GIS, specify and use XML based formats for data exchange, storage and visualisation. As examples for XML based formats with relevance to the geospatial community following languages are listed:

- Geography Markup Language (GML)
- Scalable Vector Graphics (SVG)
- Keyhole Markup Language (KML).

The *Geography Markup Language* (GML) is an XML grammar for the modelling, transport and storage of geographic feature information including the geometry and properties of the feature (OGC, 2003). According to the original GML specification, a geographical feature is defined as an abstraction of real world phenomena, associated with a location relative to the earth (OGC, 2007). The Open Geospatial Consortium, Inc. (OGC) together with the ISO Technical Committee 211 coordinates the standardization process of GML. In the current version of GML (version 3.2.1) a number of XML schemas for representing geographical phenomena are defined. These range from simple 2D features and geometric primitives (such as point, line and polygon) to features with complex 2D and 3D geometry including topological and temporal properties. Figure 2.7 gives an idea how to encode geographic features in GML.

```
<gml:Polygon>
   <gml:outerBoundaryIs>
       <gml:LinearRing>
           <gml:coordinates>0,0 100,0 100,100 0,100 0,0</gml:coordinates>
       </gml:LinearRing>
   </gml:outerBoundaryIs>
</gml:Polygon>
<gml:Point>
   <gml:coordinates>100,200</gml:coordinates>
</gml:Point>
<gml:LineString>
   <gml:coordinates>100,200 150,300</gml:coordinates>
</gml:LineString>
```

Figure 2.7: GML encoded Polygon, Point and LineString.

Scalable Vector Graphics (SVG) allows transferring 2D vector graphics, both static and animated, over the Internet or other networks. SVG is an XML specification, which can be purely declarative or may include scripting. The World Wide Web Consortium (W3C) created this standard to overcome the drawbacks of transferring raster data or proprietary data formats. Because of the reduced bandwidth of mobile devices transmitting vector data for LBS is an adequate solution. Figure 2.8 shows the SVG encoding for two squares.

```xml
<?xml version="1.0"?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN" "http://www.w3.org/
Graphics/SVG/1.1/DTD/svg11.dtd">

<svg xmlns="http://www.w3.org/2000/svg" version="1.1" width="467"
height="462">
  <!-- This is for the red square -->
  <rect x="80" y="60" width="250" height="250" rx="20" fill="red"
         stroke="black" stroke-width="2px" />
  <!-- This is for the blue square -->
  <rect x="140" y="120" width="250" height="250" rx="40" fill="blue"
        fill-opacity="0.7" stroke="black" stroke-width="2px" />
</svg>
```

Figure 2.8: Simple SVG example, representing two squares.

The *Keyhole Markup Language* (KML) is an XML-based language schema for expressing geographic annotations and visualization on 2D maps or in 3D earth browsers. KML was developed for use with Google Earth[4]. With KML it is possible to specify a set of features like *placemarks*, images, polygons, lines, 3D models and textual descriptions for display in geo-browsers, mapping services and mobile maps. With this language it is also possible to define a view on the data, with properties such as tilt, heading or altitude. Some of the grammar in KML is similar to those in GML. For its reference system, KML uses geographic coordinates in the *World Geodetic System of 1984* (WGS84). The idea of geodetic reference systems is not supported. The KML 2.2 specification has been submitted to the OGC to become an open standard for all geo-browsers. By April 2008 the OGC approved KML as industry standard. Figure 2.9 shows a sample of a simple KML file.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<kml xmlns="http://earth.google.com/kml/2.0">
<Placemark>
  <description>New York City</description>
  <name>New York City</name>
  <Point>
    <coordinates>-74.006393,40.714172,0</coordinates>
  </Point>
</Placemark>
</kml>
```

Figure 2.9: Encoding the location of New York City as Point in KML.

Beside GML, SVG and KML, *GeoRSS* can also be mentioned as XML based format to encode location as part of a *Really Simple Syndication* (RSS) feed.

2.2.2. LBS Interoperability and Standards

Information and communication technology standards are extremely important to fulfil the potential and economic success of LBSs. A user of LBSs expects efficiency, convenience, security

[4] Google Earth (http://earth.google.com, accessed August 2008) is a geo-browser, developed by Google. It was originally named Keyhole Earth Viewer created by Keyhole, which was acquired by Google in 2004.

and traceability from service providers using their personalized information. Users expect that their LBS applications accompany them when they cross cell phone roaming boundaries, they expect billing for LBSs to be integrated in their standard billing system, and they expect to work together with other users from different providers and using different technologies. Standards for LBSs are necessary because LBSs are depending on different technologies solving different tasks. In many cases service technologies and data are provided by many different companies. The trend is toward diverse technology, services and content providers, responsible only for one or two links in the value chains of LBSs.

Standards for LBSs are important for the same reason they are important in other markets. They help providers deliver usable products and services while saving time and money and reducing business risks. LBS standards benefit providers in the LBS value chain by enabling the following (Schiller and Voisard, 2004):

- Increased billable utilization of carrier's spectrum and wireless network.
- Niches for providers with special products and services.
- A business case for different content providers.
- Expansion of LBS from a niche service to a mass market service.

The key standards organizations providing LBS standards are the Open Mobile Alliance (OMA) and the OGC. OGC's main area of interest in the wireless communication industry is LBS. OGC's work involves a broad and complex set of geospatial software interoperability issues at a level in the communication technology stack. The task of the OGC in this area is to show the standardisation partners of the telecommunication industry geospatial interoperability issues such as coordinate transformations, web mapping and XML encoding of spatial information. Figure 2.10 describes the LBS standard framework from OGC and how their departments related to *OpenLS* or *OpenGIS* work together with other standards from other standardisation organisations.

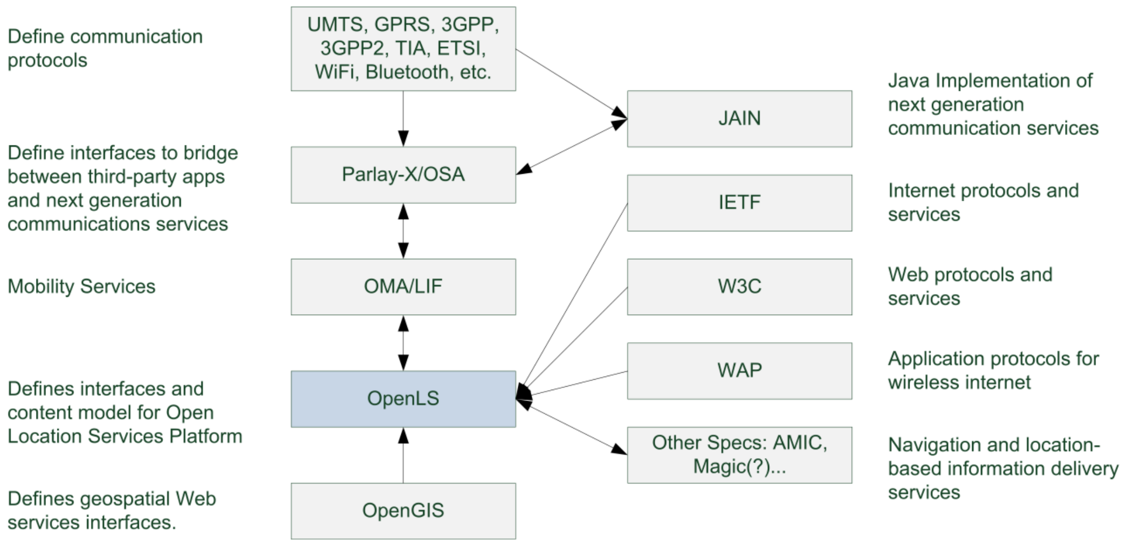

Figure 2.10: The LBS standard framework (Schiller and Voisard, 2004).

In OGC's *OpenLS* activities, OGC members have cooperatively developed the *GeoMobility Server* (GMS), a set of specifications for open interfaces and schemas that support LBSs. These standards are necessary if there has to be communication of location and time, route, types of service, etc.

across diverse technology platforms, application domains, classes of products, carrier networks and national regions. The *OpenLS* specification is a description of these core services via Abstract Data Types (ADTs). They are used to formulate a response or for the communication within core services.

2.3. Decision Analysis

Whether aware of it or not, spatial decision making is an everyday process for almost everyone. Choosing to walk to the next train station on a certain sidewalk rather than another one is a typical example. This decision is normally made ad hoc, without any formal analysis. A lot of spatial decisions are made this way, and they are often based on heuristics and internalized preferences of decision options (Morris and Jankowski, 2000). The cost of a decision at this level will be described as trivial, where factors like steepness or security of the path are considered. But spatial decision problems can become unmanageable and critical as well, which leads to an active research area for spatial decision analysis and Multiple Criteria Decision Making (MCDM). MCDM models assist decision makers by evaluating multiple choice alternatives using multiple decision criteria (Morris and Jankowski, 2000).

Decision-making is therefore an essential element of our lives and can also be critical for business success, because many natural phenomena and socio-economic activities take place in a spatial context. Spatial decision making is dependent or influenced by geographical information, but geographical information is considered as semi-structured or ill-structured, and could be multi-dimensional including spatial and non-spatial aspects. With a single step it is very difficult to model all aspects of a spatial problem. Spatial decision making usually involves a large number of alternative solutions and these alternatives need to be managed in a complex system, which supports the decision maker to select a solution from competitive alternatives.

2.3.1. Decision Problems

Simon's (1960) work on structured versus unstructured decision problems has been an underlying concept of DSSs. It provides the foundation for the classification of decision problems, including spatial decision problems (Densham, 1991, Malczewski, 1999). Any decision problem falls into the range of completely structured to unstructured decisions (Figure 2.11). For structured decisions people involved in the problem-solving process are able to identify and structure all involved elements of the problem on the basis of the relevant theory. The problems are repetitive and routine and therefore it is possible to implement computational procedures to solve structured problems without the participation of decision makers. On the other side there are unstructured decisions. Unstructured decisions occur when decision makers are unable to structure the problem on basis of a relevant theory. These decisions cannot be defined completely and are nonprogrammable. Therefore unstructured decisions must be solved by people without assistance from a computer. Unstructured decision problems are solved by the experience of the decision maker.

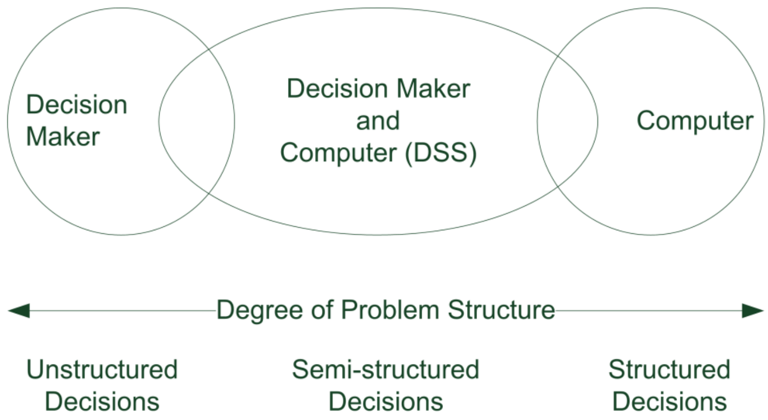

Figure 2.11: Degree of decision problems (Malczewski, 1999).

In real-world spatial decision making experts normally are not working either on truly structured or unstructured decision problems. For most decision situations spatial decision problems are indeed complex and ill structured. Most real-world spatial decision problems can be found somewhere between the extreme cases of unstructured and structured decisions. Figure 2.11 indicates decisions which are not structured and not unstructured as semi-structured decisions. This is the area where major applications of DSSs are assigned to. Semi-structured problems can be solved by decision makers with interaction of a computer-based system. The computer will support the decision maker for the structured parts of the overall decision problem.

2.3.2. DECISION MAKING PROCESS

Decision making is a process which can be divided into different phases and roles. In the overall process of decision making an SDSS can be included. The real world situation is normally the basis and starting point for a decision making process. Planning policies are the direct results of the decision making based on the objective situations in the real world. From the real world situation investigations, experiments and monitoring are done for decision making. After the decision making, implementation of the system and evaluation of the results, planning strategies and policies can be derived. Figure 2.12 draws a decision making process where an SDSS is included.

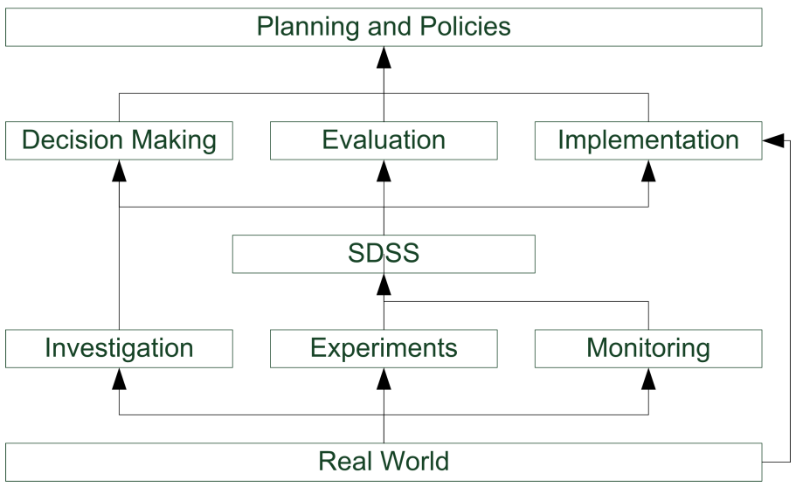

Figure 2.12: The role of SDSS in decision making.

The decision making process is actually a constantly repeated and improved dynamic process, where the goal is to enhance the decision solution. There are a number of frameworks for analysis of the decision process. Simon (1960) suggested one of the most widely accepted way for analyzing human decision making processes. In this framework the process is distinguished into three major phases:

- Intelligence phase
- Design phase
- Choice phase

In this approach the phases do not follow necessarily a linear path from intelligence, to design, to choice. At any point in the decision making process, it may be necessary to loop back to an earlier phase and reconsider this phase. Figure 2.13 shows the three phases of the decision process and also the order and loops for reconsideration.

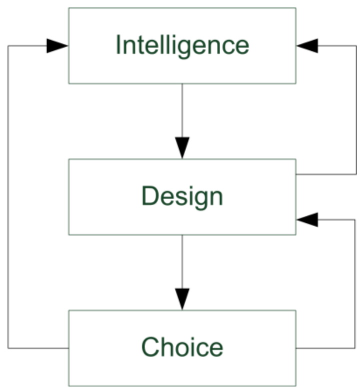

Figure 2.13: Three phases of decision making process (Malczewski, 1999).

In the *intelligence* phase, a situation is examined and a decision problem is identified. A spatial decision problem is the difference between the desired and existing state of a real-world geographic system (Malczewski 1999). The general question for the intelligence phases is about a problem or opportunity for change. The *design* phase involves the development of possible or alternative solutions to the decision problem identified in the intelligence phase. Alternatives may involve actions which will change the current situation of the real-world geographic system. The main task in the design phase is to identify reasonable alternatives. In the *choice* phase, a decision-maker or a group of decision-makers evaluate the alternatives in relation to others in term of a specified decision rule. The rule depends of the decision-maker and is used to order the alternatives under consideration and choose the best decision alternative.

In the context of decision problems related to space, Malczewski (1999) examines the potential for applying spatially enabled methods in Simon's decision phases. While the intelligence and design activities can mostly be covered by multi-purpose spatial analysis methods, the choice phase requires specific methods still missing in most GISs. Figure 2.14 illustrates the decision process based on Simon's (1960) framework for a spatial decision problem. The creation of the model is subdivided into 2 parts covering non-spatial and spatial aspects of the decision problem. Later in the process both aspects are integrated in the result to form scenarios for decision making.

Model building is critical in decision making as it clarifies the thinking of the decision maker and improves the decision-making quality. Spatial modelling techniques are used for finding relationships among geographic features and helps decision-makers to address the spatial problem clearly and logically. Spatial aspects and non-spatial aspects can be coexistent in a spatial problem, so that it is important to consider both aspects at the same time. It is difficult to model a complex spatial problem in a single step, but it is possible to model one aspect of a complex problem at a time (Hilton, 2007).

Currently a spatial decision making process is proposed by synthesising ideas for decision making processes (Simon, 1960) and the multi-criteria evaluation (Malczewski, 1999). This process includes also techniques from spatial modelling, scenario management and knowledge management. Hilton (2007) defines the process of decision making with nine specific steps:

- Problem identification
- Problem modelling
- Model instantiation
- Model execution
- Model integration
- Scenario instantiation
- Scenario execution
- Scenario evaluation
- Decision making

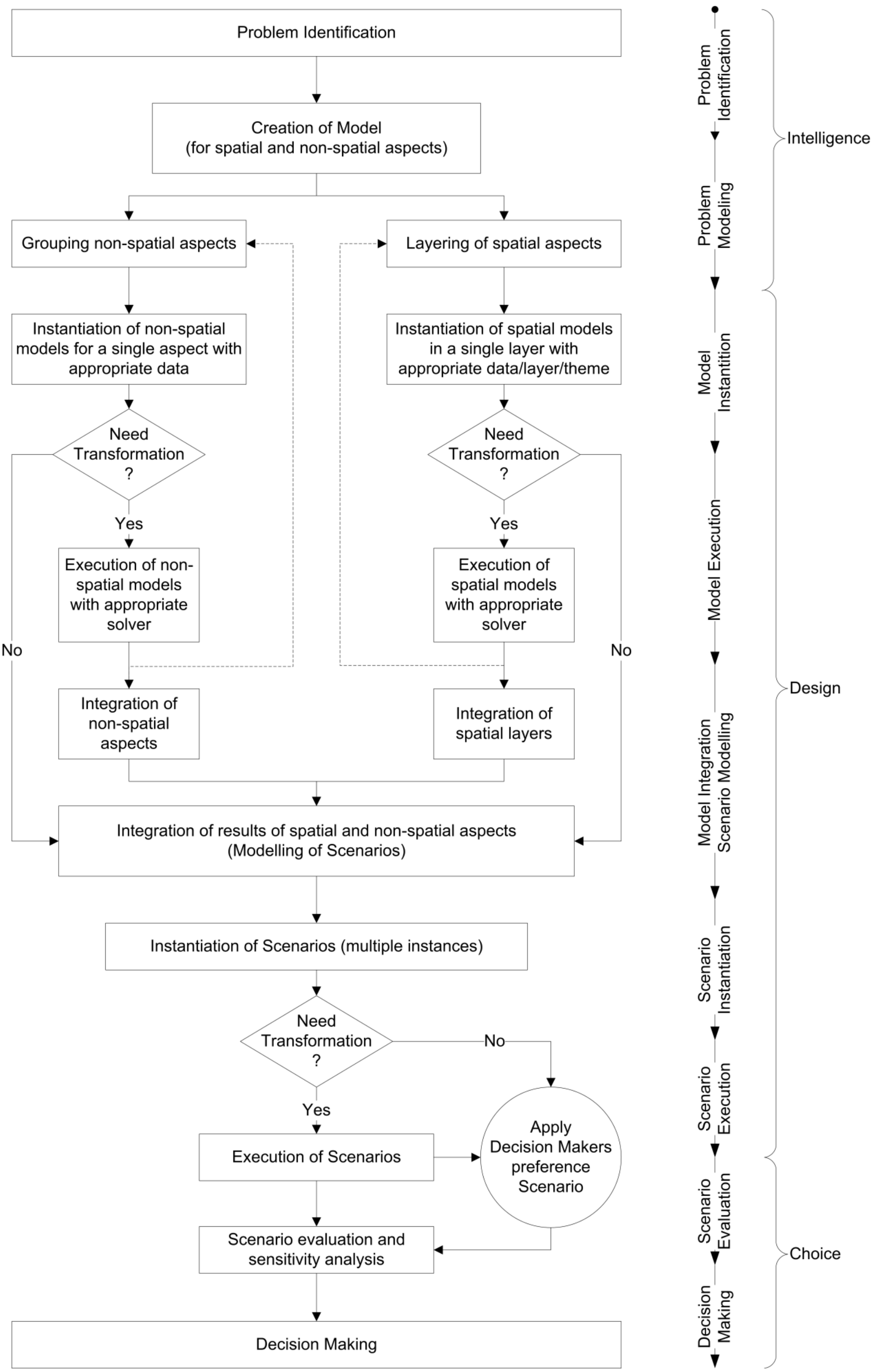

Figure 2.14: Decision making process (Hilton, 2007).

In this definition the decision making process begins with the recognition of a real-world problem that involves searching the decision environment and indentifying comprehensive objectives that reflect all concerns relevant to a decision problem. The problem is then put into a model by specifying the relevant attributes and behaviours. The parameters in a model structure are instantiated with appropriate data. Decision makers select a solver for execution of a model instance and generate a result, that is, the scenario. A scenario improves cognition by organising many different bits of information. Multiple scenarios are needed to explore different ways of seeing problems and enhancing the decision making quality. The scenario evaluation process evaluates many competitive alternatives simultaneously and helps to identify the best solution (Hilton, 2007).

2.3.3. DECISION SUPPORT STRATEGIES

In general spatial decisions are not made under respect of one criterion but multiple criteria are taken into account. Therefore, Multi-Criteria Decision Analysis (MCDA) is a sophisticated method to support the decision maker who is faced with making numerous and conflicting evaluations. Vineke (1992) describes MCDA with the "*aim to give the decision maker some tools on order to enable him to advance in solving a decision problem where several, often contradictory, points of view must be taken into account*". MCDA tries to highlighting these conflicts and deriving a way to come to a compromise in a transparent process. The key feature of a MCDM model is their capacity to adequately represent the complex nature of the decision-making process. In general, six components of MCDM problems can be listed according to Malczewski (1999):

- A goal or set of goals the decision maker aims to achieve;
- The decision maker or group of decision makers involved in the decision process along with their preferences with respect to evaluation criteria;
- A set of evaluation criteria (objectives and/or attributes) on the basis of which the decision maker evaluates alternative courses of action;
- The set of decision alternatives, that is, the decision of action variables;
- The set of uncontrollable variables or states of nature;
- And the set of outcomes or consequences associated with each alternative-attribute pair.

These elements and their relationships are represented in Figure 2.15. In this figure the decision matrix, consisting of a set of columns and rows, is derived from attributes and alternatives.

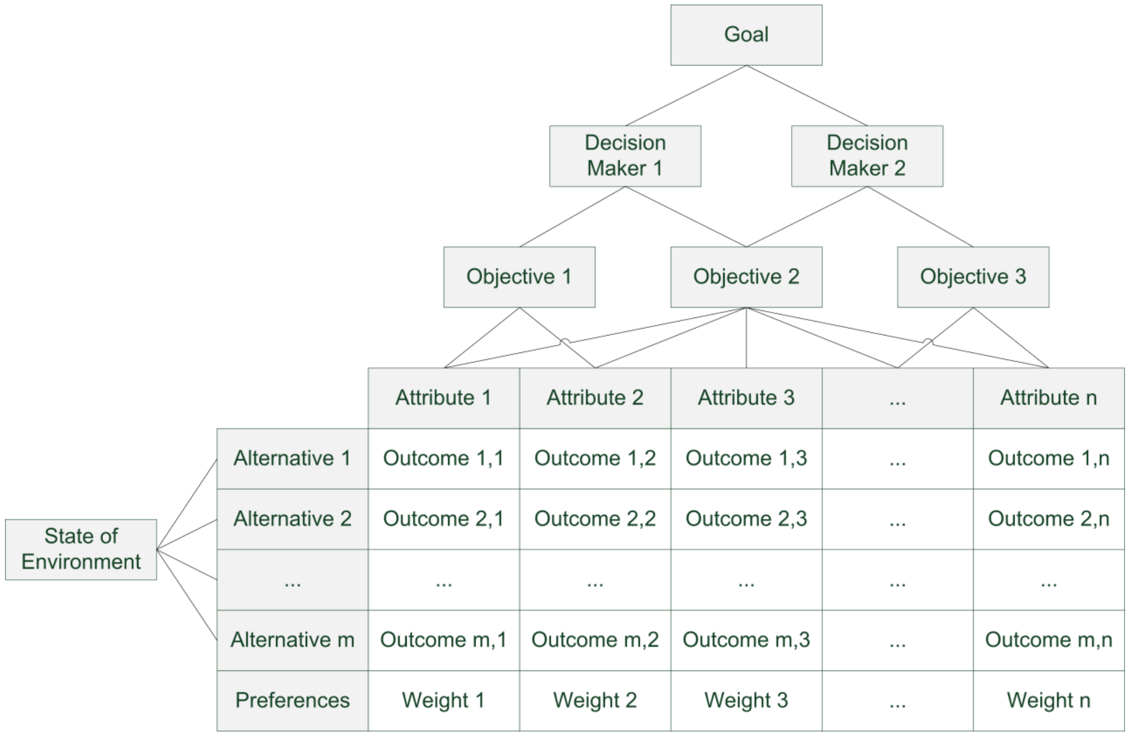

Figure 2.15: Framework for multi-criteria decision analysis (Malczewski, 1999).

In the decision matrix of Figure 2.15 the structure of the columns consists of levels representing the decision makers, their preferences, and evaluation criteria, organized in a hierarchical structure. The rows of the decision matrix represent decision alternatives. These are influenced by the nature of environment and are described as uncontrollable factors of the decision problem. Such uncontrollable factors could be weather conditions, land coverage or an action of a competitor.

The basic classification of multi-criteria decision problems is done in:

- Multi-objective decision making (MODM)
- Multi-attributive decision making (MADM)

This distinction between MODM and MADM is based on the classification on evaluation criteria into attributes and objectives. Each the two approaches can be further subdivided with other classifications (Malczewski, 1999).

2.4. SPATIAL DECISION SUPPORT SYSTEMS

The area of SDSS is an important subset of DSS because of the need of a spatial component for decision making (Keenan, 2007). The authors of Crossland and Wynne (1994) presented the empirical evidence of the usefulness of a spatial approach to decision making. Some authors argued for incorporation of GIS in decision support areas more than 15 years ago (Eom *et al.*, 1993, Keenan, 1995).

2.4.1. DEFINITION

The branch of DSS is an important area of information systems research. A DSS is broadly defined as a computer-based information system that affects or is intended to affect the way people make decisions (Silver, 1991, Hilton, 2007). A major limitation of these systems, applied for managerial decision making, is the inability to exploit spatial and temporal data. Because of the lack for handling spatial data a new type of DSS has emerged, known as *Spatial Decision Support System* (Sugumaran and Sugumaran, 2005). An SDSS is an interactive, computer-based system designed to support a user or group of users in achieving a higher effectiveness of decision making while solving a semi-structured spatial decision problem (Peterson, 1998). The effectiveness of decision making is achieved by incorporating the decision maker judgments and computer-based programs into the decision-making process. Decision support means that the system helps the users to explore the decision problem in an interactive and recursive fashion in all phases of the decision-making process.

Spatial Problems, which can be solved by SDSS, are normally categorized into:

- allocation,
- location,
- routing and
- layout problems.

Also GISs have been described as SDSSs per se, or as generators for more specific SDSSs (Maguire, 1991, Rinner, 2003, Keenan, 2007). Mostly an SDSS incorporates both GIS functionalities such as spatial data management, cartographic display, etc., as well as analytical modelling capabilities, a flexible user interface, and complex spatial data structures (Goodchild, 2000). Thus, an SDSS provide a framework for integrating

- analytical and spatial modelling capabilities,
- spatial and non-spatial data management,
- domain knowledge,
- spatial display capabilities and
- reporting capabilities (Armstrong and Densham, 1990).

Typically, SDSSs are flexible integrated systems build built on a GIS platform combining different modules. Similar to DSSs, SDSSs support *what if* analysis and is designed to communicate and help the user to understand the results (Goodchild, 2000). On a conceptual level, Malczewski (1999) defines the necessary components of an SDSS in the narrower sense as:

- geographic database
- model base
- dialog (user interface)

Traditional GIS-based SDSSs are complex systems that require sophisticated infrastructure and resources in hardware and capital. Even systems with client server architecture tend to use thick clients that require lots of resources (Sugumaran and Sugumaran, 2005). Some of popular models used for an SDSS are:

- Multi-criteria evaluation (MCE) models

- Network optimization models
- Ordered Weighted Averaging (OWA)
- Artificial neural networks
- Spatial regression
- Spatial clustering

The main characteristics of spatial decision problems include:

- a large number of decision alternatives,
- the outcomes or consequences of the decision alternatives are spatially variable,
- each alternative is evaluated on the basis of multiple criteria,
- some of the criteria may be qualitative while others may be quantitative,
- there is one decision maker or a group of decision makers involved in the decision-making process,
- the decision makers have different preferences with respect to the relative importance of evaluation criteria and decision consequences, and
- decisions are often surrounded by uncertainty.

Based on these facts provided above and despite an ongoing discussion, there is no agreement on a single definition for SDSSs. And this explains why most authors' simple ends up with a list of characteristics of what an SDSS might compromise. The lack of a widely accepted and unambiguous definition for SDSSs does not limit the number of systems referred as SDSS rather enhance the variation of different applications. Keenan (2007) identifies three different groups of decision makers for SDSSs:

- decision makers coming from the traditional area of GIS applications like geology, forestry, land planning, etc,
- decision makers for routing and location analysis and
- decision makers from areas where the importance of location is neglected at the presence or the awareness for location is currently rising, e.g., marketing.

Multi-Criteria Spatial Decision Support Systems (MC-SDSSs) can be viewed as a part of a broader field of SDSSs. The need for using such systems is derived from situations where complex spatial problems are ill-defined or semi-structured, and decision makers cannot define their problem or fully articulate their objectives. The decision making process adopted to solve semi-structured spatial problems is often perceived as unsatisfactory by decision makers. MC-SDSS tools offer unique capabilities for automating, managing, and analyzing single-user and collaborative spatial decision problems with large sets of feasible alternatives and multiple conflicting evaluation criteria (Ascough *et al.*, 2002). As technology progresses, there is increasing opportunity to use an SDSS in a variety of domains. Flexible support of decision-making processes to solve complex, semi-structured or unstructured spatial problems can offer advantages to individuals and whole organizations. An important step is to synthesize ideas, frameworks, and architectures from GIS, DSS and SDSS. In addition, concepts from spatial modelling, model life cycle management, scenario life cycle management, knowledge management, and MCDM methodology are explored and leveraged in the implementation of flexible spatial decision support system using object oriented methodology and technology (Hilton, 2007).

2.4.2. HISTORY OF DECISION SUPPORT

For approximately 40 years information technology researchers have designed and investigated DSSs. Efforts in this discipline started with model-driven DSSs in the late 1960s, theory and developments in the 1970s and implementation of spreadsheet DSSs and group DSSs in the early and mid 1980s. Business intelligence evolved in the late 1980s and early 1990s and nowadays researchers focus on knowledge-driven DSSs and implementation of web-based DSSs and mobile DSSs.

With the development and availability of minicomputers, timeshare operating systems and distributed computing computerized decision support became practical. The implementation of such systems began in the mid-1960s, but on the various perceptions of people different milestones in the history of decision support where highlighted and considered as important (Power, 2004). As technology evolved new computerized decision support applications were developed and researcher used multiple frameworks to help and understand these systems. Power (2007) organized the history of DSS into five DSS categories, including: communications-driven DSSs, data-driven DSSs, document-driven DSSs, knowledge-driven DSSs and model-driven DSSs. The study of decision support systems is an applied discipline that uses knowledge and theory from other disciplines. For this reason geospatial research questions have been examined by GIS experts who were building SDSSs.

The idea of a model-driven spatial decision support system (SDSS) evolved in the mid and late 1980's (Armstrong *et al.*, 1986). Data-driven spatial DSSs are also common (Power, 2007). A model-driven SDSS emphasizes access to and manipulation of financial models, optimization and simulation models of spatial phenomena. Model-driven DSSs use limited data and parameters provided by decision makers to support them in analyzing a situation. In general, a data-driven SDSS emphasizes access to and manipulation of a set of spatial data. Research from SDSS originated from two different sources:

- decision support,
- geographic information science.

The integration of these two approaches has resulted in SDSS, which combines the decision analytic power of traditional DSS and the spatial capabilities of GIS. A schematic representation of the progress in SDSS development is shown in Figure 2.16. This figure shows the development of SDSS both form the development of geographic information science and form the decision support discipline.

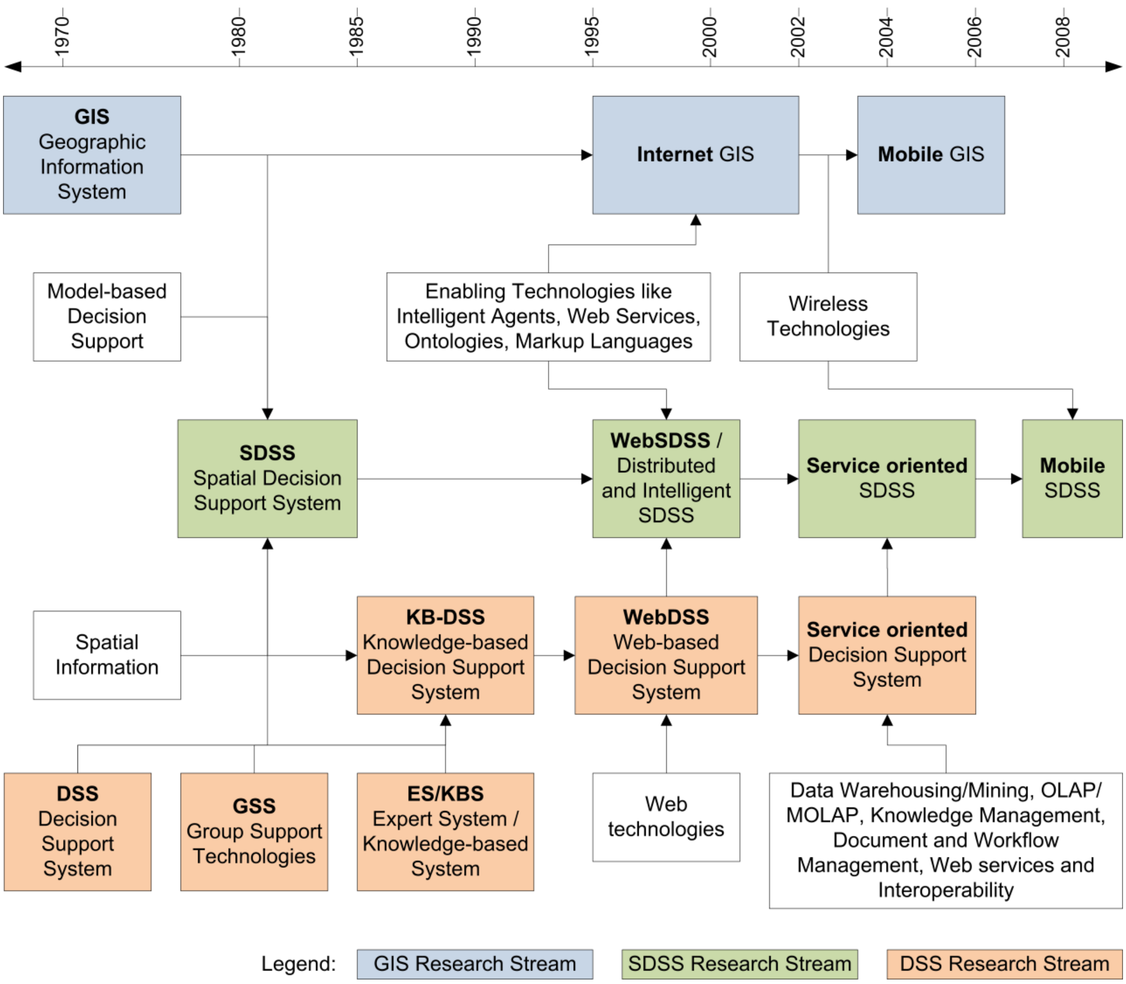

Figure 2.16: Progression of SDSS development, adapted from Sugumaran and Sugumaran (2005).

GISs evolved in the past four decades, but only recently it can be noticed that GIS technology incorporates with mainstream information technology decision support solutions. Beside traditional application fields of GIS there is a growing interest to use geospatial technology for decision support within the business community. Strengths for analytical and visualization capabilities are the decisive factor that organizations are adopting their GIS solutions for decision support to gain competitive advantage. While the first generation of GIS provides some modelling capabilities, they were inadequate for supporting any type of business decision support. During this time, considerable efforts were made in designing and developing DSSs by the information systems community. These model-based and knowledge-based approaches for building DSSs were adopted by the GIS community. This marked an important step in GIS driven evolution path and SDSSs were created (Sugumaran and Sugumaran, 2005). Huerta *et al.* (2005) and Jarupathirun and Zahedi (2005) provide a review and summery about GIS based SDSSs and their applications within the business domain.

The most recent phase in the GIS-driven development has been influenced by the enormous growth of the Internet. Internet based technologies has been used for integrating GIS and DSS functionality within web applications and services. Several researchers have demonstrated the use of the Internet for spatial decision support (Rinner and Malczewski, 2002, Rinner, 2003, Sugumaran *et al.*, 2003).

2.4.3. APPLICATIONS OF MC-SDSS

Historical and recent applications and studies illustrate the wide range of spatial multi-criteria analysis to real-world decision problems. MC-SDSS have been applied for a variety of situation including solid waste management (Caruso *et al.*, 1994), nuclear waste disposal facility location (Carver, 1991), land use planning (Diamond and Wright, 1988), land suitability and use analysis (Eastman *et al.*, 1995), health care resource allocation (Ewart, 1994, Jankowski and Eward, 1996), land resource planning (Faber *et al.*, 1995), land use analysis (Fischer *et al.*, 1996), routing in transportation risk assessment (Ganter and Smith, 1995), agriculture land use (Janssen and Rietveld, 1990), water resource management (Lotov *et al.*, 1997, Martin *et al.*, 1999) or land development (Wu, 1998). These examples include different types of MC-SDSS and MCDA methods. A more detailed list of MC-SDSS can be found in Malczewski (1999) on pages 336-337.

2.4.4. GEOGRAPHIC INFORMATION SYSTEMS AND DECISION SUPPORT

A Geographic Information System (GIS) is defined as computer system to acquire, store, retrieve, analyze and visualize spatial data (Longely *et al.*, 2005). GIS is one area of information systems which has expanded enormous recently and has become a mainstream information system. The growth of GIS has been driven by the importance of spatially related data (Keenan, 2007). Most GIS today can only be used for tasks, described in the definition, and there is a lack in focusing on effective decision-making capabilities. SDSS is developed for the purpose to support the decision-making process. Keenan (2007) has advert to the importance to examine the relationship of GIS and DSS, and described the integration of a DSS model in a GIS software with the example of ArcView[5]. A comparison between GIS and SDSS is done in Table 2.3. In most cases, the decision maker expects SDSSs to provide support in two aspects (Yan *et al.*, 1999):

- One is to help them detect the problems, enhance their cognition to the problems, find out about the essentialities and focus their main objectives and tasks.
- The other is to assist them to generate different alternatives to a problem, compare these alternatives and finally select the most suitable one.

According to specific problems and various levels of details decision makers have different requirements to SDSS.

[5] ArcView is a desktop GIS software system developed by ESRI.

Table 2.3: Comparison between GIS and SDSS (Yan *et al.*, 1999).

Content of comparison	GIS	SDSS
Focus	Information	Decision making
Object oriented	Well-defined problems	Ill-defined problems
Function	Describe the real world and detect the problems	Support decision making in order to exploit and remould the real world
Characteristics	Processing routinely, inflexible	Processing according to requirements, flexible
Result Output	Maps and tables	Decision making schemes and best solution
Status of user	Positive, system driven	Active, user driven
Scope of application	Management at middle level	Decision making at high level
Objective	Improve working efficiency, seek for quickness	Improve decision making ability, seek for effectiveness

Depending on the processes, requirements and characteristics of spatial decision making, developers of such systems should adopt guiding ideologies in designing and building SDSS software tools and applications. This will improve the availability of decision makers to make scientific decisions related to their discipline. Because of different characteristics and the different focus of SDSS tools compared with a GIS the design and architecture could be completely different between those systems. In addition GISs can be modified and used as basis to add decision support functionality.

During the period of growth in the importance of GIS also decision support became more prominent and was applied in many different fields. The trend of greater use of spatial information has lead to the integration of mapping functionality in software available on a mass market basis. Different sources of spatial information are combined to derive decisions. New technologies and internet trends allow combining different kinds of spatial and non-spatial information (mash-ups) which are the core for decision support. While additional map display functionality is introduced in popular software- and web applications or services, it is unlikely to provide the full range of spatial query operations associated with a fully fledged GIS. However, this mass market use of mapping and GIS products creates a further extension of the demand for spatial data. Decision makers, who make use of basic mapping products, are likely to become aware of the need for more sophisticated software or systems that enable them to make adequate decisions. Most decision makers and users of such systems come from fields that are not traditional users of GIS technology. If even a small proportion of users of mainstream business applications were attracted to using geographic data, it would represent a large increase in the market of GIS software and data (Keenan, 2007).

GIS as tool for data collection is very useful and allows the storage of data in a spatial database based on standards and well-defined principles. For the SDSS in this case data can be accessed from the spatial database and different criteria can be formed from the provided database. This allows experts for SDSSs to focus on strategies for data extraction in regard to the decision problem. A good introduction about spatial databases, working on spatial (e.g. point, line polygon) and non-spatial (e.g. attributes) data is given by Rigaux *et. al.* (2001) and Shekhar and Chawla (2003). In developing an SDSS an important task is designing and building the relevant geographic databases for a specific application and making these operational through data integration. Data from various sources have to be integrated in a spatial database or linkages created between them.

GIS are often used explicit or implicit to support spatial decisions in different fields of applications. In some cases GIS is defined as DSS (Maguire, 1991) or used as decision support generators (Keenan, 2007). Many GI based systems are described as being DSSs on the basis that the GIS assisted in the collection or organization of data used by the decision maker, but do not include all components of a DSS. The GIS capabilities for supporting spatial decisions can be analyzed in the context of decision-making processes (Malczewski, 1999). Decision-making processes can be adapted for the integration of GIS functionality (compare section 2.3.2).

2.4.5. WEB-BASED SDSS

While a conventional SDSS is used in managing critical decisions the number of users or decision makers is limited. Web-based SDSS are being developed to provide decision support facilities related to geographical information and analysis to a larger audience through the Web. The objective in the design process of a web-based DSS is the combination of appropriate data and models for decision support using a web browser. Different authors provide status reports about web technologies providing decision support services over the internet or examples for web-bases decision support services or web enabled decision support products.

Power (2003) has suggested that the domain of discourse of DSS has expended to such an extent that the traditional boundary of DSS has become fuzzy and is blurring with related technologies such as business intelligence, Online Analytical Processing (OLAP), data warehousing, knowledge management, and web services.

The client-server model used in designing web GIS applications enables users to gain access to a spatial database through remote procedure calls and open database connectivity. Since network computing evolved and the Internet provides the infrastructure for peer-to-peer computing, a next phase for web-based GIS applications has started, including the progression of mobile GIS environments (Sugumaran and Sugumaran, 2005). This architecture permits many-to-many communications which can be applied for distributed spatial problem solving. SDSS functionalities can be modularized and implemented as components or services in such Service Oriented Architectures (SOA). The overall system can be build from these modules or the modules can be embedded into other applications. A service-based SDSS provides ubiquitous access to spatial computational services from anywhere and anytime using any device. These components actually act as spatial web services and users can compose a set of these services to archive support in a particular decision problem. Web service technology is supported by several key protocols and standards such as XML, JSON, REST, WSDL, SOAP and UDDI. One further advantage of SOA for SDSS consists of minimizing the cognitive load on end users because of its ability to deal with heterogeneity in hardware as well as software components that may be written using different languages. It provides interoperability by considering needed translations between different components and services working together (Sugumaran and Sugumaran, 2005).

TRADITIONAL WEB SDSS ARCHITECTURES

Web SDSS includes a web-based geographic information system as a problem solver and facilitates geographic data retrieval, display and analysis. It combines several different components including web based user interfaces, Internet interface programs, computational models and spatial- and non-spatial databases or data services (Sugumaran and Sugumaran, 2005).

Form the client-server approach there are two different ways to set up a web-based SDSS:

- Server side processing system
- Client side processing system.

With the server side approach all major processing and computational tasks are performed on server side only the representation of the results and interaction with the user is done on the client side. The client side architecture uses a thick client where some GIS or processing functionality is included. The geographic data is accessed from a server and preloaded on the client machine where further processing is done. Components of a more sophisticated server-side web SDSS are shown in Figure 2.17. Components like a knowledge server, GIS server, decision support server, databases server and web server are included on the server side. The GIS server covers spatial data retrieval, data transformations and spatial analysis tasks, while the decision support server includes spatial and non-spatial models to generate and rank decision alternatives. The database server hold geographic and attribute data relevant for the decision problem and the knowledge server tries to represent expert information in the field of application. Finally, the web server is responsible for the communication between all components and the client.

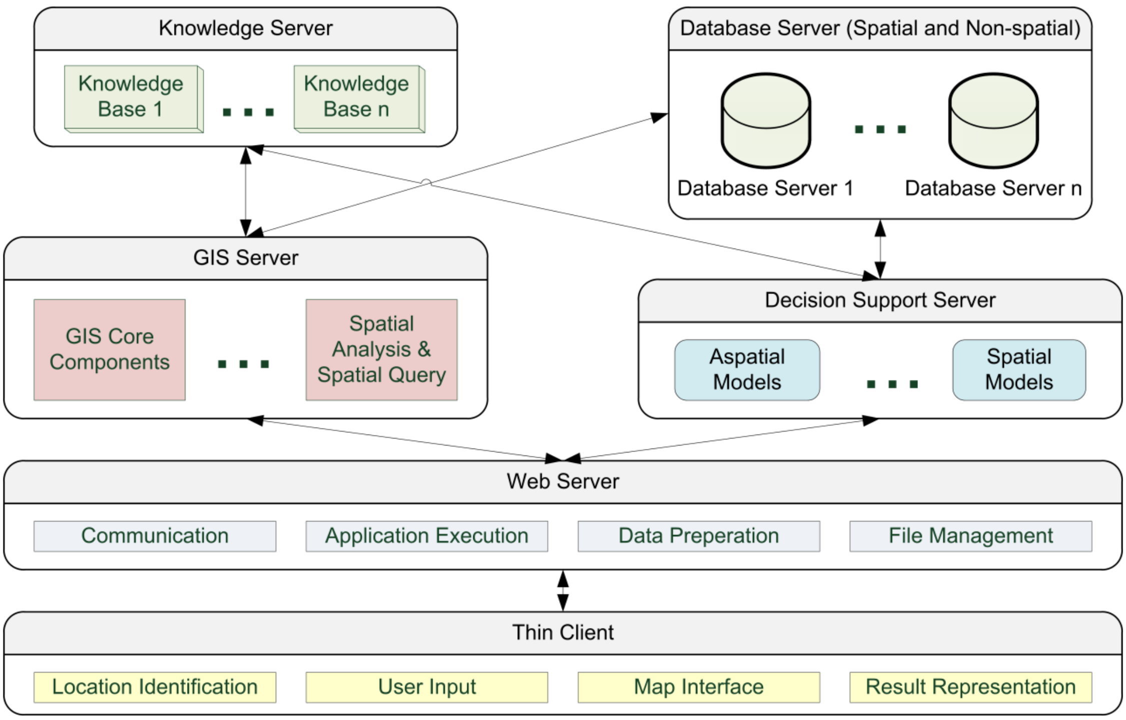

Figure 2.17: Schematic representation of web SDSS components for server-side approach, adopted from Sugumaran and Sugumaran (2005).

TRENDS FOR WEB SDSS

The development of the web has also influenced multi-criteria spatial decision support. The widespread connectivity between computers over the Internet has facilitated the storage and retrieval of information, included spatial data. Recently spatial data become easily manageable and publishable over the Internet. There have been numerous efforts to develop the web as a tool for sharing geographic information, through clearinghouses, digital spatial data libraries, etc. Currently efforts are taken in the area of intelligent geographic information management.

Modularization of code is a recent trend likely to affect the world of multi-criteria spatial decision support. Existing spatial toolsets will be radically affected by current trends toward open and interoperable systems. In order to make SDSSs interoperable, it will be necessary to establish common standards of meaning, so that concepts and models in one system can be linked to concepts and models in another. Data models and database structures used in SDSSs become more complex and varied to represent the diversity of geographic phenomena in our world with more accurate models and to improve decision support. Metadata plays an important role for choosing and understanding available data, and how these data should be accessed and integrated in a decision support model. The standardization process for geospatial metadata will go beyond basic vector and raster structures. All these efforts will increasingly provide the means for interoperability of data between different platforms, systems and software packages (Sugumaran and Sugumaran, 2005).

Another trend for multi-criteria decision support technologies is the demand of a spatial-temporal decision support system, rather than only a spatial system. According to Sugumaran and Sugumaran (2005), in the future it will be necessary to:

- *Develop methods for handling massive spatial data sets:* As geo-referenced data sets become larger, experts must develop methods to use, classify, and manipulate the rich information inherent in very large (i.e., global) spatial data bases.
- *Analyze spatial and space-time data:* Decision makers must extend exploratory methods of analyzing spatial data to include space-time data so that we can develop models (decision and otherwise) that better represent reality.
- *Develop computationally intensive tools and methodologies:* Computationally intensive tools can allow more effective use of large data sets, more sophisticated and extensive simulations of complex spatial phenomena, and the solution of complex location and distribution problems.

Another keyword is collaborative decision making where the interaction of a group of decision maker to solve a decision problem is supported by the system. Service oriented architectures for DSS are mainly used for collaborative decision making and are referred as service web-based collaborative decision support services (Wang and Cheng, 2006).

3. METHODOLOGY

"All truths are easy to understand once they are discovered; the point is to discover them."

Galileo Galilei

This chapter emphasises on background about spatial decision support for location based services. Questions like, why and how should a user be supported in spatial problems with a mobile device, are discussed. The chapter *Methodology* includes a definition for location based decision services and gives examples for applications. Technologies, which may influence the design of location-based decision services and the architecture of such systems is described. The last part of this chapter shows an overview about SDSS models for mobile applications.

3.1. LOCATION BASED SERVICE AND DECISION SUPPORT

The combination between LBSs and Decision Support seems reasonable, since people decide about spatial problems when and where these problems became relevant. The disciplines of LBS and Decision Support are discussed in the theoretical part of the work. Tremendous benefits may be achieved from the widespread adoption of LBDSs, providing large segments of the population with real-time decision support for purposes ranging from trivial, e.g., way-finding services, to critical, e.g., emergency response (Bäumer *et al.*, 2007).

Personalisation is seen as one important trend for LBSs. Multi-criteria decision analysis enhances location-based queries to a more personalized level. This is the case, because the user can set the own preferences in form of criteria and weights on these criteria which decisively effects the decision support results. When personalized data is used privacy is an important concern. Using weighted criteria to enhance personalization demand a responsible use and transfer of information.

First prototypes, already introduced, enhance the way of location-based queries beyond standard single-attribute solutions through an approach that gathers user preferences in a qualitative way (Rinner and Raubal, 2005, Raubal, 2006). These qualitative preferences were used as input for MCDA (Malczewski, 1999). It allows the user to take several relevant attributes into account, which are considered during the decision-making process and let users assign weights to define their relative importance. Currently there are less human subject tests done to evaluate whether multi-criteria decision support for mobile devices is applicable or is accepted by the user and how the communication between the human and the system works (Rinner *et al.*, 2005, Bäumer *et al.*, 2007). Uran and Janssen (2003) describe some reasons for the lack of success of decision support

systems in general. One reason described in their work is that users find the system too detailed, time consuming and costly to use. Other reasons are related to the general complexity of the systems, while still others are related to the uncertainty of the model output and on the appropriateness for solving the decision question. For LBDS applications these reason are even more relevant under acceptance that unlike ordinary SDSSs used by experts, LBDSs are used by natives. In the planning and development phases of LBDSs these drawbacks have to be considered.

3.1.1. LOCATION BASED DECISION SERVICES

LBDSs assist people in their decisions while they move through the physical environment. To a certain degree, LBS support decision-making by answering spatial queries and provide the user additional information, which can be relevant for the decision process. Most LBSs use spatial queries, like "find the nearest POI from my current location" or "find the shortest path from my current location to a gas station". Some of the applications use a combination between spatial and attributive queries, e.g., "find the nearest hotel which costs less than a certain amount of money". However, spatial multi-criteria methods go beyond querying by enabling users to evaluate and rank decision alternatives based on preferences and the combination of multiple criteria. LBDS systems explicitly use decision support methods to suggest decision alternatives based on user preferences, the combination of multiple criteria and the current location.

3.1.2. APPLICATIONS FOR LBDSS

Applications for LBDSs can be found where people have to decide about a problem related to space, immediately on the current location. Such near real-time location related decisions can be supported by a LBDS application operating on a mobile device. Possible application areas include:

- tourism and leisure,
- transport and logistics,
- emergency management and emergency response, and
- personalized mobile advertising.

Raubal and Rinner (2004) introduced a mobile hotel finder application that uses MCDA principles. Users get a list of decision-relevant attributes to be used as evaluation criteria. On the interface of the mobile device the user identifies good, fair, and poor criterion scores or ranges for comparison of standardized criterion scores and defines the relative importance of criteria by assigning weights. The weighted criterion scores are then combined based on a decision rule, resulting in an evaluation score for each decision alternative. Rinner and Raubal (2005) extend the hotel finder application using the Ordered Weighted Averaging (OWA) decision rule that allows users to specify a personal decision strategy as part of their decision-related preferences. OWA defines a continuum from optimistic to pessimistic decision strategies in a mathematical sense and uses a second set of weights to emphasize high or low standardized criterion scores. The implementation on the mobile client is realised as ArcPad[6] add-on. In a bar finder variant developed by Rinner *et al.* (2005), alternative user interface designs for location-based MCDA were developed to simplify

[6] ESRI ArcPad is software for mobile GIS and field mapping applications using handheld and mobile devices.

mobile decision-making processes. Figure 3.1 shows a screenshot of the *BarChoice* application running on a Personal Digital Assistant (PDA).

Figure 3.1: Screenshot of *BarChoice* application showing an interface for weighting criteria through predefined classes (Rinner *et al.*, 2005).

Emergency management involves phases like mitigation or prevention, preparedness, response, and recovery. GIS and related methods and tools, including positioning technology (e.g., GPS) and remote sensing imagery, are being used to various extents in those phases (Cutter, 2003). For example, spatial analysis methods can help with hazard identification and risk assessment in the mitigation phase. During the response to an emergency event, GIS is often used as mapping tool to support rescue and recovery operations. Emergency management is seen as an important application field for desktop-based SDSSs and systems for different types of disasters have been designed (Saltf and Dunsmore, 2000, Castle and Longley, 2005, Erden and Coskun, 2007, Wang and Cheng, 2007). While some desktop-based SDSSs include sophisticated mathematical models for emergency management and response, the use of mobile GIS in this area has been limited to map creation to support field operations. An extension of mobile GIS with explicit decision

support functionality, e.g., MCDA, is proposed and illustrated with an emergency response scenario by some authors (Cai *et al.*, 2004, Rinner, 2007).

The authors of Cai *et al.'s* (Cai *et al.*, 2004) have described a hypothetical scenario of geo-collaborative crisis management that can be used to demonstrate mobile decision support functionality: After a hurricane hit the coast of Florida, a first responder team uses mobile map-based systems for the communication with the emergency operations centre to find appropriate shelter for hurricane victims. The responder team finds a group of people that need to be evacuated from the flooded region to a shelter that provides certain services and has enough capacity. The emergency operations centre compiles a map with shelter and background information that is shared with the first responders. The first responders request information on the capacity and caregiver staffing of an ad-hoc selected shelter and then decide to use that one as it "looks practical". This decision is based on the first responders' geographic intuition and facilitated by viewing the collaborative map. As an extension of this scenario, Rinner (2007) assumed that the emergency operations centre adds a shelter layer to the base map with shelter information like current capacity, staffing. Further, the emergency operations centre sends a default decision strategy to the first responders' mobile client that consists of default importance weights for decision criteria such as the travel distance to shelters and the shelter attributes. Other settings for an MCDA method such as the decision risk could also be transmitted. The first responders activate the MCDA process with a click and receive a suggested target shelter. On the map, several other shelters appear to be much closer to the first responder team's location. The first responders vary the decision strategy, for example by increasing the decision risk slightly above the suggested default. Soon, one of the nearby shelters becomes the top-ranked option and the first responder team takes measures to guide the victims to that shelter (Rinner, 2007). The screenshot of the *ShelterChoice* application in Figure 3.2 gives an impression about the user interface for the responder team.

Figure 3.2: Screenshot of *ShelterChoice* application (Rinner, 2007).

A personalized mobile advertising system could also be realized as LBDS. With a user profile interests of products can be specified. These interests and other preferences like time and distance to a shop offering products can be combined as criteria in a multi-criteria decision support analysis. According to these information advertisement alerts show up if the customer is within a certain radius of the shop, which offers products she/he is interested in.

3.2. Technologies influencing a LBDS

There are several technologies which can influence the development of a LBDS. LBDSs are internet based distributed systems and therefore most technologies which are important are related to information and web technology. Sugumaran and Sugumaran (2005) explain that, "*web*

technologies will have a profound impact on the next generation of SDSS." SDSSs in the Internet age will not be individual, interactive systems with specialized modelling features and interfaces, but will instead be combinations of data and services linked over the web for a specific purpose. SDSS tools will be modular and reusable (Casey and Austin, 2002). In this section some of these technologies, like geoweb services, ontologies and semantic web, intelligent agents as well as sensor web are described.

3.2.1. GEOSPATIAL WEB SERVICES

The W3C defines a *Web Service* as a software system designed to support interoperable machine to machine interaction over a network. Beside this definition there are a lot of other definitions, which tries to describe web services (Booth *et al.*, 2004, Mateos *et al.*, 2005). Web services can be accessed over the Internet and executed on a remote system hosting the requested services. The core technology associated with web services is the standardization of data or message exchange between applications or systems during every stage of their life cycle, including transporting, invoking, and discovering (Hilton, 2007). Web services are ideal for distributed and service oriented architectures.

The communication between client and server is based on XML messages that normally follow the Simple Object Access Protocol (SOAP) or XML-Remote Procedure Call (XML-RPC). XML provides a platform independent structured information format, which hides all underlying transport protocols. Figure 3.3 describes a client-server communication based on SOAP. Additional a web service contain a machine readable description of operations supported by the server. This description is written in the Web Services Description Language (WSDL). It allows automated client-side code generation in many Integrated Development Environments (IDEs). The developer of a web service often uses a standard registry or catalogue, such as Universal Description, Discovery and Integration (UDDI) to publish the web service. With catalogues other developers can discover registered web services. These described characteristics distinguish a web service form traditional and proprietary distributed systems, such as Distributed Common Object Model (DCOM), Java Remote Method Invocation (RMI) API or Common Object Request Broker Architecture (CORBA).

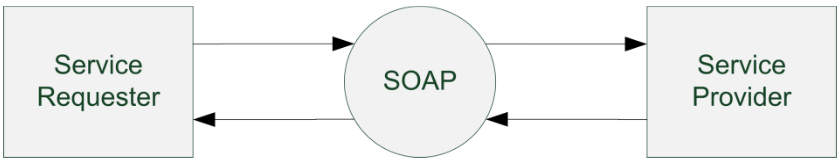

Figure 3.3: Elements of SOAP communication between requester and provider.

One of the major benefits of web services is the possibility to create interoperability. A web service hides all the details of implementation under a well-defined interface, and thus other applications or services can invoke such a web service through a defined interface. Hilton (2007) gives a detailed overview about major benefits of web services.

Web Services are a set of tools that can be used in a number of ways. The three most common types of use are:

- *Remote Procedure Call (RPC) Web Services* presents a distributed method call interface. RPC is a communication technology that allows a piece of software to cause a procedure

to execute in another address space without explicitly coding of the details for the remote interaction. RPC may be referred to as remote invocation or remote method invocation. An RPC is initiated by the client sending a request message to a known remote server. This request message executes a specified procedure using supplied parameters on the server. A response is returned to the client where the application continues along with its process. There are many variations and subtleties in various implementations, resulting in a variety of different and sometimes incompatible RPC protocols. While the server is processing the call, the client is blocked.

- *Service-oriented Architecture (SOA)* provides concepts for implementing web services. The basic communication is done via messages, rather than an operation. This is also referred to as message-oriented services. The functions or services are loosely coupled with the operating systems and programming languages underlying the applications.
- *Representational State Transfer (REST) Web Services* attempt to emulate Hypertext Transport Protocol (HTTP) and other protocols by constraining the interface to a set of well-known, standard operations. The focus is on interaction with resources, rather than messages or operations. RESTful web services can use WSDL to describe SOAP messaging over HTTP (Fielding, 2000).

Geospatial web services operate on the same technological elements as ordinary web services. Geospatial web services provide spatial data and GIS functionality via a network to client applications and users. So it can analyze, manage, distribute and operate spatial information. The advantage is that very specific GIS functionality and data can be provided to applications without using comprehensive GIS software. Further benefits of geospatial web services against file based spatial information exchange are discussed by Hilton (2007). Some authors predict that geospatial web services will revolutionize how companies use and interact with geospatial information (Gonzales, 2003, Sugumaran and Sugumaran, 2005). Figure 3.4 shows the principle architecture of web services.

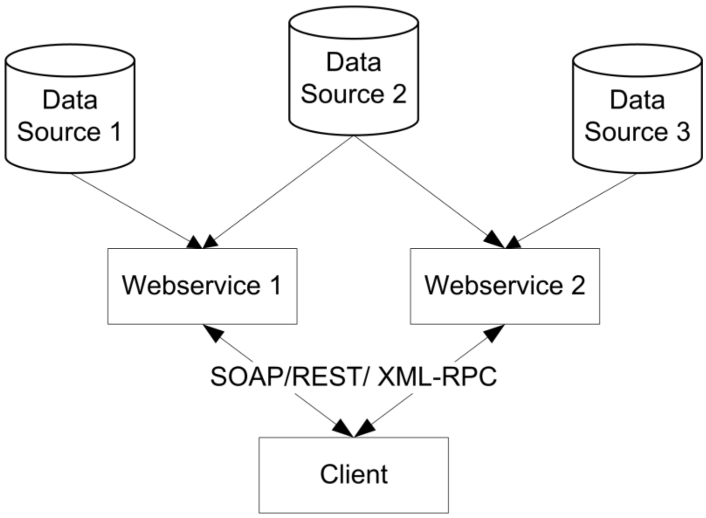

Figure 3.4: Architecture of a web service or geospatial web service.

Important players of standardization efforts for geospatial web services are ISO/TC 211 and the OGC.

3.2.2. Geospatial Web 2.0 Services for Decision Support

Web 2.0 has recently become a very popular term, which is used for describing different concepts of the Internet, where it is not clear if the term can be used in a meaningful way. In most cases it is referred as a trend for using web technology to enhance collaboration among users and information sharing. These ideas have led to the development of web-based communities such as wikis, blogs, social-networking sites and *folksonomies*. Also the geospatial web is affected by the concepts of publishing and sharing spatial information. Table 3.1 shows differences in using spatial information in the traditional web and web 2.0.

Table 3.1: Properties of the Geo-Web 1.0 and Geo-Web 2.0.

Geo-Web 1.0	Geo-Web 2.0
Static 2D maps	Dynamic 2D/3D maps, globes
File Transfer (FTP)	Direct use of web services (SOA)
Individual Websites	Web Service Mash-ups and APIs
Proprietary protocols	Standard protocols (W3C, OGC, ISO)
User hosted services	Remotely hosted services
Clearinghouse nodes	Catalog portals

The way maps and geographical information is published over the *World Wide Web* has tremendously evolved recently. A few years ago, data (also spatial data) was provided by experts in their fields and published on a server. The geospatial web was primarily based on OGC standards and architectures like Web Map Service (WMS), where maps were rendered on demand at the server, and delivered to the client as an image embedded in a web page. For any interaction or change on the map, it was necessary to request a newly rendered map image from the server (Elson *et al.*, 2007). In 2005 *Google Maps*[7] pioneered a new class of *Web Maps*. Instead of strict client/server architecture, *Google Maps* followed a distributed approach that shifted much of the application logic into the client browser (Simon, 2008). A reduction of page reloads is achieved with sophisticated *JavaScript* programming and asynchronous, background data transfer (Asynchronous JavaScript and XML or AJAX). This new approach has the unexpected side effect, that enthusiast web developers, reverse-engineered the mechanism[8] to overlay own data on the map and started to create their own applications, using their own geo-referenced and alternative data. The term *mashup* has since become a synonym for web applications build up by combining functionality and content from various sources, using a single user interface. These kind of web-based applications have gained huge popularity and therefore most web map service providers officially support these ideas by providing public Application Programming Interfaces (APIs).

The *mashup* phenomenon undoubtedly leads to an increase in public interest about geospatial technologies outside traditional academic and industry circles (Rouse *et al.*, 2007). Additional numerous web services are available, where geospatial data can be accessed and used for own applications. The term *neogeography* is used in contrast to *paleogeography* to indicate the

[7] Google Maps (http://maps.google.com, accessed in June 2008) is a Google service offering mapping technology and local business information. The Google Maps API provides a number of utilities for manipulations of maps and adding content to the map through a variety of services, allowing to create own maps applications. The documentation can be found at http://code.google.com/apis/maps (accessed in June 2008).

[8] Paul Rademacher created the first mashup (http://www.housingmaps.com).

differences between new forms of end-user driven online geography versus traditional web GIS-based approaches (Goodchild, 2008). Turner (2006) defines *neogeography* as a collection of technologies and "*tools that fall outside the realm of traditional GIS*" which "*combines the complex techniques of cartography and GIS and places them within the reach of developers.*"

The definitions about *neogeography* are still vague and the tools, standards, and interfaces that form its technological basis are fragmented (Simon, 2008). In order to harmonize the various technologies involved, the *neogeography* community is working on the specification of the *GeoStack* – an agreed reference model that covers the entire life-cycle of geospatial data on the web, from capturing to consumption (Turner, 2006). Figure 3.5 shows a draft of the *GeoStack*, which includes the phases create and capture, produce and publish, aggregate and consume. Each of the four phases of the life-cycle uses specific technologies and interfaces or exchange formats.

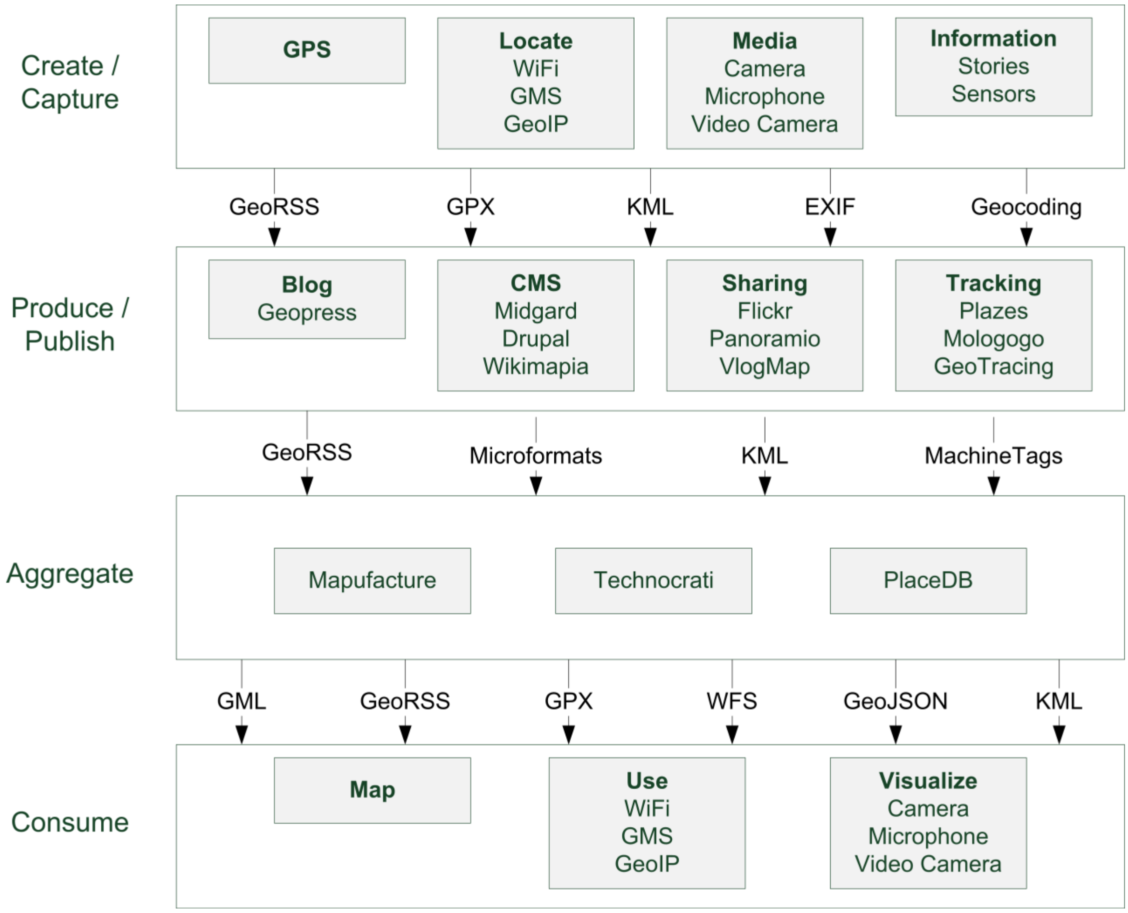

Figure 3.5: *GeoStack* adopted from Turner (2006).

Data from *Web 2.0* services are created and provided by the community itself. Such data can represent enormous value for decision support. In many cases people have to face the same or similar problems and find a decision. The experience someone has made for a certain decision problem is extremely valuable for someone else facing the same or a similar situation. For this reason data recorded by someone having experience in a decision problem can be integrated in a decision support system. For example, someone is trying to find an adequate hotel in a foreign city. The visitor can find a decision based on characteristics of the hotel room, like price, location, size, included breakfast etc., but he has no idea about cleanness of the room, noise or quality of the breakfast. Here the information of another visitor, who knows the hotel, could help in her/his

decision finding process. All this information is somehow integrated in a subjective rating of the hotel. Therefore a LBDS which integrates community rating in the decision model might be useful for the decision maker. The *web 2.0* gives everyone the possibility of quickly accessing and processing large, spatially distributed geospatial datasets in a way that SDSSs make use of them.

3.2.3. ONTOLOGIES AND SEMANTIC WEB

The philosophical meaning of ontology is a particular system of categories reflecting a specific view of the world. It is a study of conceptions of reality and the nature of being or existence and forms the basic subject matter of metaphysics. Guarino and Giaretta (1995) tries to clarify the terminological issues about ontologies, including the difference between Ontologies (uppercase) and ontologies (lowercase). "*Ontology as an engineering artefact describes a certain reality with a specific vocabulary using a set of assumptions regarding the intended meaning of the vocabulary words*" (Fonseca and Egenhofer, 1999). In the geographic world an object oriented view of the environments is used frequently. This means that ontologies for geographic information systems focus on object technology and the nature of spatial and non-spatial objects. Ontologies can play a major role in the design and development of GIS-based systems, because this concept allows the establishment of correspondences and interrelations among different spatial entities (Sugumaran and Sugumaran, 2005). Experts speak of Ontology-Driven Information Systems (ODISs) when an explicit ontology plays a central role in the system's life cycle. Ontologies can be integrated to enhance the interoperability between GISs (Fonseca and Egenhofer, 1999), and avoid inconsistencies between build in GIS elements and user expectations of the application.

Ontologies capture the model of knowledge for a particular domain. They make it possible to describe resources on the web and relationships between those resources. For example, a geospatial ontology could include geographical and topological concepts like location, adjacency, containment and navigation. A web page may contain a postal mailing address, written in textual form. Without ontologies this information has no meaning to an agent. If the address is marked-up with instances of a geospatial ontology, an agent is able to infer an individual geographic location both in absolute terms (coordinates) and relative terms (country, county, neighbourhood, etc.) (Casey and Austin, 2002).

The decision maker uses a DSS normally to generate a list of decision alternatives for comparison and evaluation of different objectives. In order for DSSs to carry out the well-described functions, the future vision for such systems is that they will rely on metadata, which describes attributes, objectives, context and constraints and will therefore be ontology-driven. In this area further research is necessary to develop ontologies and tools to encode semantics for decision making.

The semantic web is an evolving extension of the current web in which semantics of information and services is defined, with the goal to understand and satisfy the request of peoples and machines to use the web content. The core of the semantic web lies in the ability to express meaning and establish relationships between resources and to process queries based on these relationships. Methodologies borrowed from artificial intelligence, knowledge representation, and other domain specific communities will achieve this goal (Casey and Austin, 2002).

An Internet based SDSS combined with semantic web methodologies allows the interpretation of spatial data and geospatial web services using agent technologies to extend the capabilities of distributed SDSSs. For example, a semantic web-enabled navigational SDSS can use ontological references to provide directions to the user for navigation based on the criteria and constraints

defined by the user. Physical and semantic constraints can work together for reasoning and computational effectiveness (Malyankar, 1999, Sugumaran and Sugumaran, 2005). Research in the development of the future web is focused on the exchange of not just documents, but on the exchange of data and services, human-to-machine communication and machine-to-machine communication. This sort of information exchange is based on the ability to embed meaning or semantics in conjunction with data and services. In order to reduce the impact of manual interaction to web-based SDSSs new semantic methodologies are necessary (Casey and Austin, 2002).

3.2.4. Intelligent Agents

Several researchers suggest a cooperation between agent technology and decision making problems (Manson, 2000, Sengupta and Bennett, 2003, Sugumaran and Sugumaran, 2003). In some works agent-oriented model frameworks are suggested to overcome some of the limitations of traditional SDSS approaches. It can be assumed that these efforts are just the beginning using agent technology and GIS, for making SDSSs more applicable for spatial problem solving.

In a service oriented approach for SDSSs where different combinations of data and services are linked over the web, intelligent agents are capable of inferring knowledge form these models and supporting decisions. Intelligent agents for SDSSs must be able to query distributed loosely structured data and dynamically return decision alternatives, which are in line with the decision maker's objectives and preferences. This is an enormous challenge because agents lack in human cognitive abilities to interpret spatial and non-spatial data. For example, agents cannot easily read and understand maps. For this reason efforts have to be taken to develop encoding mechanisms to integrate geospatial semantics, and provide spatial data in an machine-readable form to support spatial reasoning (Casey and Austin, 2002).

3.2.5. Sensor Web

The *Sensor Web* is a network of spatially distributed sensors combined with geospatial functions to monitor and control the environment. Distributed sensor platforms (pods) communicating with each other are forming a sensor network. The architecture allows every pod to know what is going on with every other pod throughout the sensor web at each measurement cycle. The novelty of the sensor web architecture lies in the interaction of each individual piece and to act and coordinate as a whole. By definition, a sensor web is a web-centric, open, interconnected, intelligent and dynamic network of sensors to collect data, and fuse and distribute information (Tao *et al.*, 2004). The term sensor web is sometimes used to refer to sensors connected to the Internet and sometimes used in conjunction with projects of the OGC. The OGC architecture is described by Botts *et al.* (2007).

Cities and major facilities such as airports are full of sensors and security cameras, which are accessible or can be made accessible via the Internet. This is an important source for disaster managers to support their decisions in terms of a crisis. Multiple interoperable technologies could be used together to support actions during an emergency event. The information can be used for decision support in a central disaster management office or also direct near the location of the event. Using mobile technologies such as location based services based on sensor data as decision support can help relief units in their coordination. Technologies like functions as access to map images and raster and vector data, new specifications for Sensor Web Enablement (SWE), Building

Information Models (BIMs), Geospatial Digital Rights Management (GeoDRM), and service chaining are addressed to build a system for sensor based mobile decision support. OGC's sensor web enablement specifications make it possible to find and control online sensors as diverse as radiation counters, anemometers, security cameras and NASA imaging satellites (Botts *et al.*, 2007).

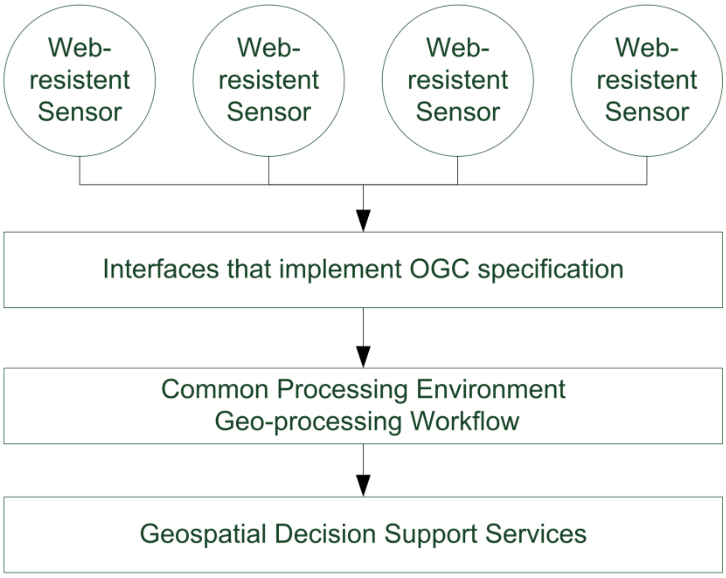

Figure 3.6: Diverse kinds of sensors can be found on the web, evaluated, and controlled, sometimes in a thoroughly automated fashion.

One of the most obvious information from a sensor which might be integrated in a LBDS system is weather information. The current weather situation of a specific location might be important for deciding to go to there or not.

3.3. ARCHITECTURE FOR LBDS

In software engineering and web based systems *multi-tier* architectures or *n-tier* architectures are applied for client-server communications, with more than one distinct software agent. The idea of an n-tier architecture is to structure different conceptional aspects of a software system in different layers. The communication between layers is restricted to adjacent tiers. Beside two-tier and five-tier architectures the most widespread use of multi-tier architectures refers to three-tier architectures.

3.3.1. THREE-TIER ARCHITECTURE

Three-tier is a client-server architecture which includes different tiers for user interface, business logic and data access. Each tier can be developed and maintained as independent modules and can run on separate and different platforms. The three-tier model is considered to be software architecture and a software design. Apart from the usual advantages for modular software with well defined communication interfaces, the three-tier architecture allows independent upgrades

and changes in requirements and technology for each tier. Figure 3.7 illustrates all layers of the three-tier architecture.

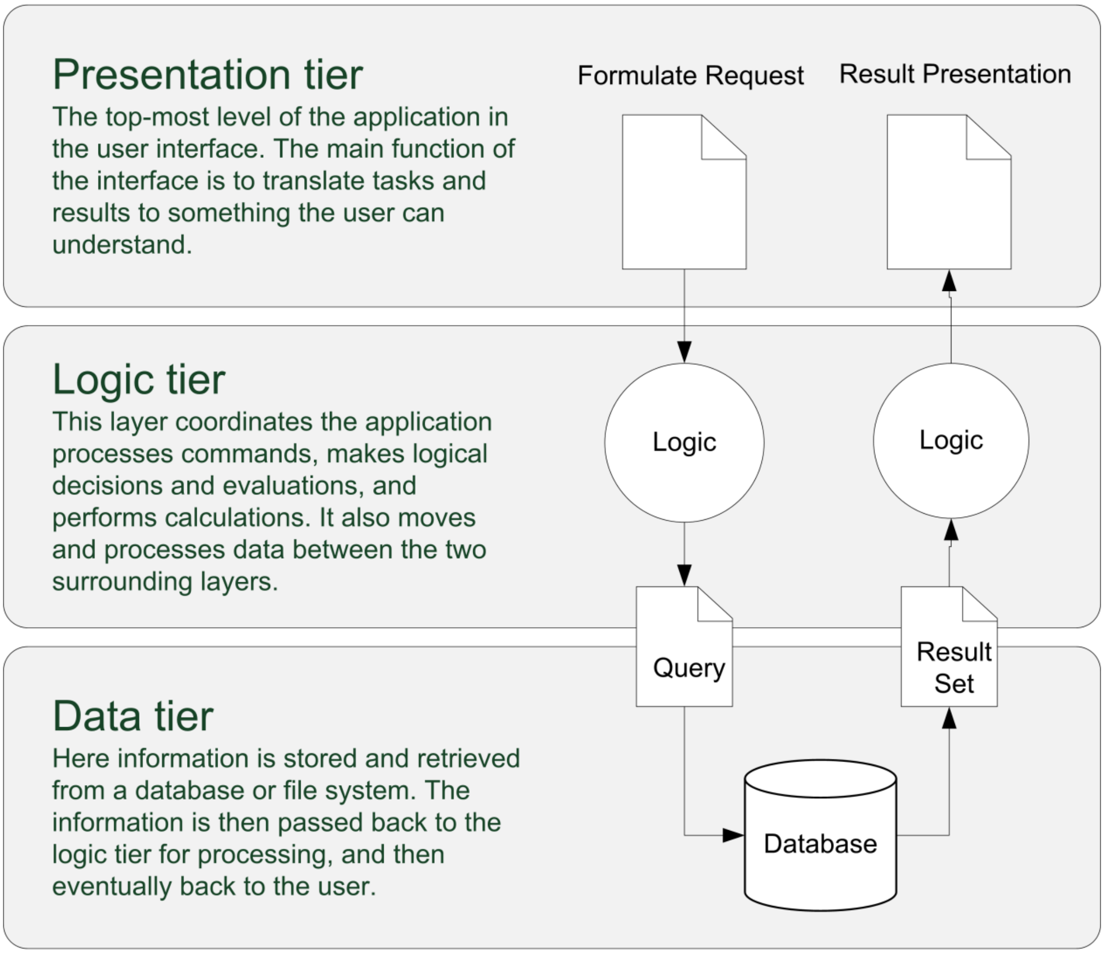

Figure 3.7: Three-tier architecture.

Typically, the user interface runs on a client, e.g., mobile device, the logic tier consists of one or more individual modules running on an application server and database management systems are located on a database server. The three-tier architecture consists of the following layers:

- The *Presentation Tier* is the topmost level of the application. The presentation tier displays information related to services in the logic tier. It communicates with other tiers by outputting results to the client. The main task of the presentation tier is the communication with the user.
- The *Application Tier* or *Business Logic* is pulled out from the presentation tier and controls an application's functionality by performing detailed processing. The middle tier may be multi-tiered itself.
- The *Data Tier* consists of database servers where information is stored and retrieved. This tier keeps data neutral and independent from application servers or business logic. Maintaining data on its own tier also improves scalability and performance.

3.3.2. LBDS Systems as Three-tier Architecture

For SDSSs there are two kinds of systems according to the system architecture: stand-alone SDSS and network SDSS (Yan *et al.*, 1999). The stand-alone systems have an SDSS installed on a single computer while the network systems may have the different parts of the SDSS distributed, which can be accessed as services over a network. Network SDSSs may have different architectures like a client-server structure or SOA. A mobile SDSS could be implemented as a standalone application on the mobile device. Here a snapshot of a geographic dataset together with decision models and a user interface are installed on the device. The whole decision support process is performed by the software on the device without connecting to any network. Some factors suggest conceiving LBSs to use network structures to provide results for the user. For this reason only network based SDSSs are considered in this work. Spatial and non-spatial data form the basis of the decision support process and needs to be as up-to-date as possible. Therefore, it seems reasonable to request needed data over a network.

As defined before an SDSS consists of database, model-base and user interface. The model-base represents the business logic of the application. Normally, the database is located on server side and the management of the data and model and user interaction is located on client side. There are applications where the model-base is implemented on the client side (Rinner, 2005), while other application uses business logic from server side. Figure 3.8 shows the difference of these two approaches.

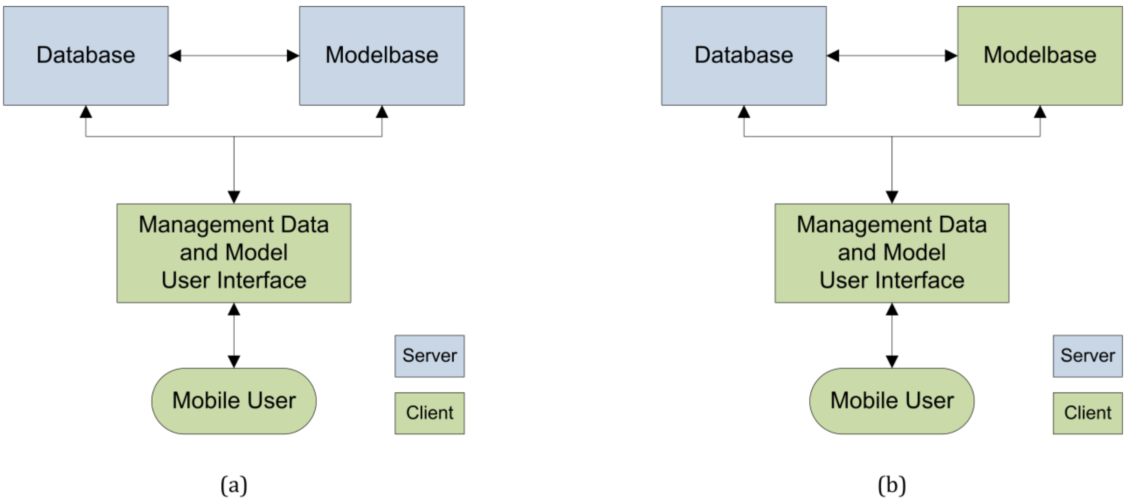

Figure 3.8: Components of a network SDSS. In (a) database and model base are on the server side, where in (b) only the database is on the server side.

A LBDS system is considered as information system and it is obvious to design a LBDS application with a multi-tier architecture. The distribution of the components of an LBDS could vary according to the objective of the application. Figure 3.9 shows a three-tier architecture for LBDS systems.

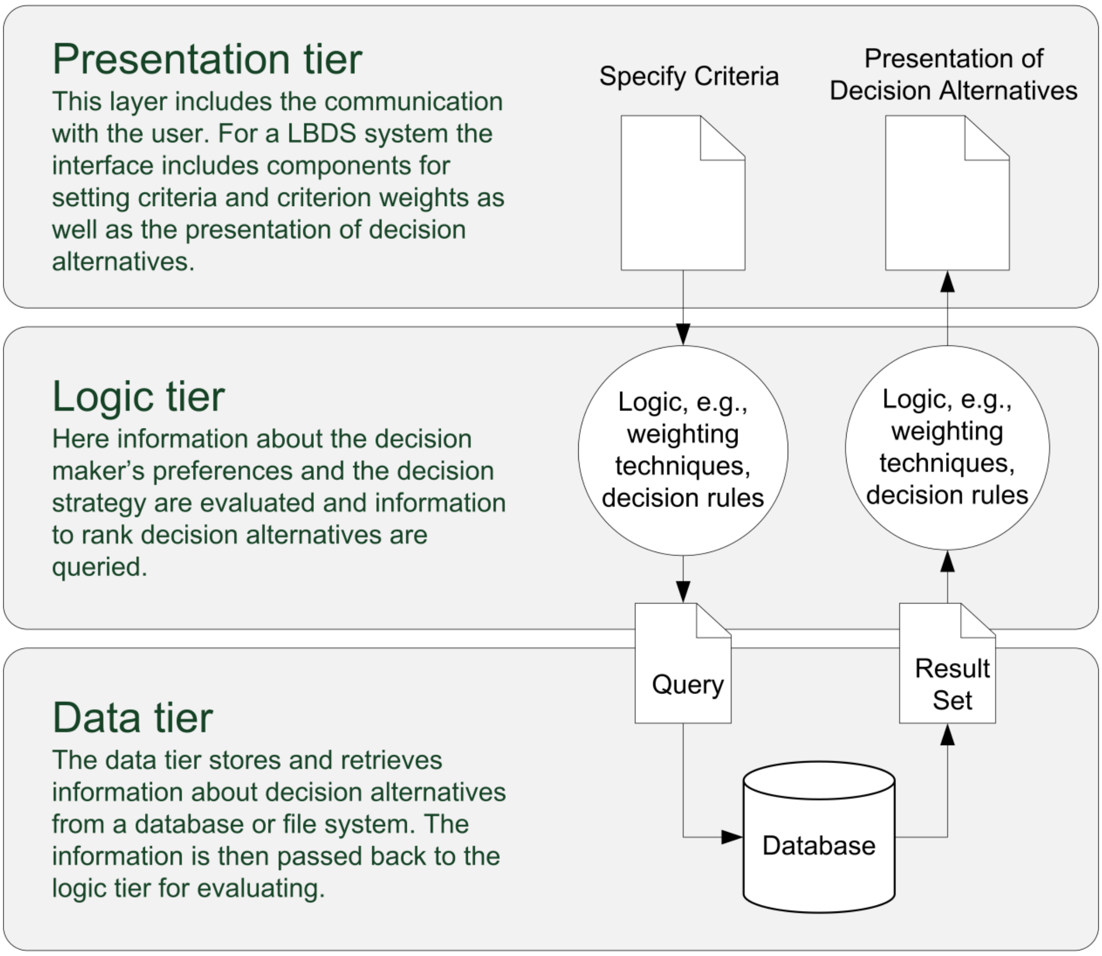

Figure 3.9: Three-tier architecture for LBDS systems.

Implementing the database and model base on the server side and the user interface on the mobile client allows keeping the client side slim and the whole system is more flexible if well-defined interfaces and standards for the communication between the components are used. The database can consist of various and distributed sources like base-maps for visualization and content relevant geographic data and attributive information or community generated content to form decision alternatives. Another data source may come from sensors, which are connected with the Internet. Examples for such sensors are weather sensors or sensors measuring air pollution.

The model component, located in the logic tier, consists of data pre-processing and MCDA techniques or other optimization techniques. In the step of data pre-processing geographical data is prepared for the actual decision support process. For the preparation ordinary GIS functionality like geo-computation and GIS analysis is used. After the preparation of geographic data it could be the case that decision support methods are only dealing with textual or numerical data rather than spatial data. The multi-criteria methods contained in a model base of an mSDSS can be mathematically as simple as a weighted averaging (Rinner and Raubal, 2005). But depending on the problem and expected accuracy of the result it can become more complex. The user interacts with the model setting different parameters influencing the model or decision strategies. These parameters have to be transferred from the user interface to the model-base, while other parameters may be stored on the server itself.

The User Interface (UI) of an LBDS is naturally located on the client side. Basically the UI of an LBDS provide functionality to represent geographical data via a map or in textual description. Additionally to this representation there is the deed for a dialog component to interact with the user on the evaluation criteria, such as selecting decision criteria and setting criterion weights. Rinner *et. al.* (2005) discuss UI design for mSDSSs based on a common mobile GIS package. The work on the UI design is a critical part for the overall system because SDSSs are dealing with semi-structured decision problems and the interaction with the decision maker is essential to solve the decision problem efficiently. The difference between the UI design of mSDSSs and GIS systems is the interaction of parameters with the geographic results in consideration of all limitations of the mobile device. For this reason the simplification of the UI is an important task. Another component of the UI that has been identified as a characteristic for a DSS is a report generator (Densham, 1991). For a mobile decision support systems this does not appear to be necessary, because it can be assumed that decisions are made right after the suggestions while moving (Rinner and Raubal, 2005). A report generator enhances the traceability of the decision support process and is often used by experts for group-decision making. Therefore, it is necessary that the UI of a LBDS include functionality to represent geographic data and the decision results as well as interaction and review of the decision parameters.

3.3.3. COUPLING OF COMPONENTS

LBSs are a special form of web services dealing with location of the user or mobile device (Schiller and Voisard, 2004). Basically the actual geographic location of the device is combined with information delivered by a network, normally the Internet. As other web services, LBSs can be implemented as client-server model. The server component provides information and processes the requests from the client. The client side interacts with the user and takes user inputs and returns results to the user. There the graphical user interface is implemented. Mobile devices, which are limited in their functionality, are the client side for LBSs. These limitations include processing power, display resolution and other factors. Therefore it should be considered which parts of an SDSS can or should be integrated on the server side and which functionality can be realized on the client side. An SDSS consist of three main parts, as discussed before, including the data, model and user interface which are interacting with each other. The coupling or system integration between the model, UI and data in general is a major issue in developing an SDSS (Jun, 2000). In most cases data processing tasks will be handled by the server and the UI part will be implemented on the client. There are three approaches, which handles the integration of the components of an SDSS differently:

The *loose coupling* strategy combines the capabilities of separate models for GIS functions and decision support methods (e.g., MCDM) by manual data transfer. To work on a spatial decision problem the experts have to switch between GIS software and DSS software (MCDM software). Figure 3.10 shows the two different models for GIS and DSS. The user has to access these different models separately. The implementation of an SDSS using loose coupling of the components can be done with data exchange. The exchange of one model does not affect the other model since the data exchange remains on the same agreement. The disadvantage using loose coupling is the user interaction because of the need to handle two separate models in separate software systems.

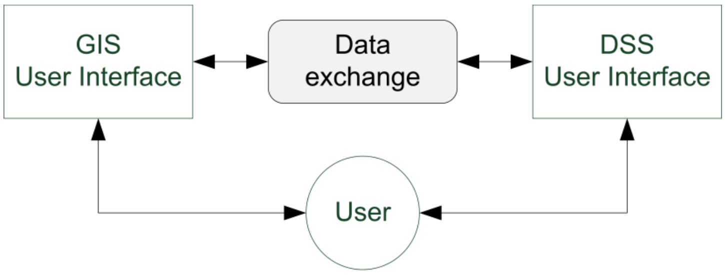

Figure 3.10: Loose Coupling.

The *tight* or *close* integration strategy is based on a single data or model manager and a common UI. With this strategy there is no need to switch between different software systems to run (multi-criteria) decision analysis. An example for this strategy is IDRISIs[9] Decision Analysis tool. The advantage is that the user can work on a single UI which interacts with the models.

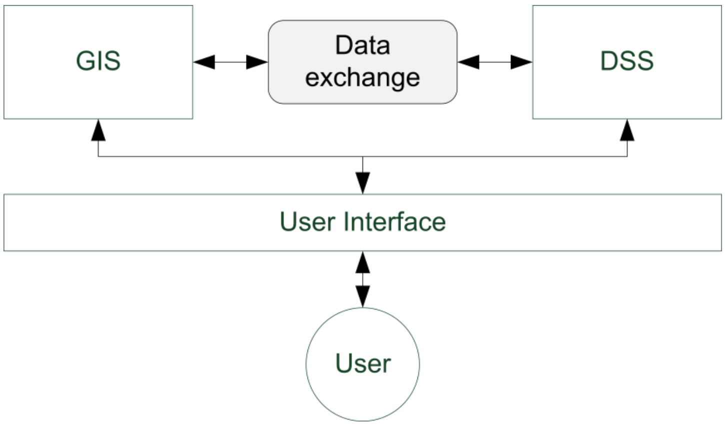

Figure 3.11: Tight Coupling.

Interoperable is the ability of two or more software components to directly cooperate or communicate despite of their differences in programming languages, interfaces (communication boundary between two entities) and execution platforms. For interoperable integration the ability to use the same data model and set of methods is important. It is no integration but coupling of components if only well defined communication interfaces for data exchange without a common set of methods are used.

[9] Clark Labs IDRISI is a GIS for the analysis and display of digital spatial information. Since May 2007 it is on version 15 also known as IDRISI Andes. IDRISI is specialized for raster data and popular as an academic tool for teaching the principle theories behind GIS in colleges and universities. IDRISI has special packages for spatial decision support (http://www.clarklabs.org, accessed June 2008).

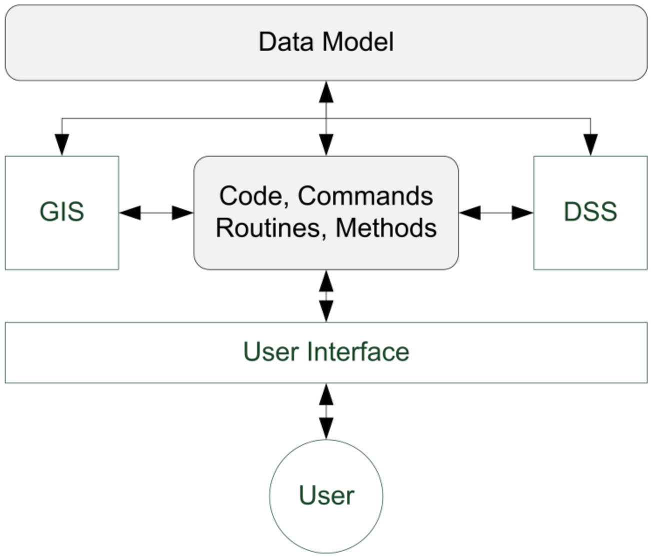

Figure 3.12: Interoperable Integration.

The interoperable integration of components is very useful for a LBDS. It allows that such systems evolve from a closed expert oriented technology to an open user-oriented technology.

GIS and decision making would likely combine interoperable geo-processing services that can be chained to build specific SDSSs. GIS and decision making will move to distributed systems where everyone has access to use it. This includes the distribution of geospatial data as well as geo-processing or computational services. SDSSs can be mashed up over the internet like it is known with data services.

Some LBDSs implement the model component on the client side (Rinner, 2005) but it can be also considered to implement the model-base on the server to reduce effort for the client. A server side design minimize the processing effort on the client while maximize server load and accessibility. If there are additional sub-models, they could be handled with a hierarchical approach (Malczewski, 1999).

3.4. SPATIAL DECISION SUPPORT METHODS FOR MOBILE APPLICATIONS

Spatial decision support methods are responsible to interpret user preferences and find a decision support strategy. Composed methods and models form the model-base of an SDSS, the main differences between SDSS and GIS.

Models can be used to analyse and evaluate the particular application area, to make preliminary processing of the data, and to make forecasting about future trends. It is through model analysis that people can abstract the complex real world. Based on the abstracted knowledge, people can make clear objectives, and plan initial schemes for the problems. Each scheme can be evaluated by other models to show its benefit and malpractice. Comparing the evaluation results of all the schemes, decision makers can make a choice among the schemes. If all the schemes are not fitting

for the decision problem, the decision maker can change the analysis models and even add new schemes to repeat the process again (Qiao *et al.*, 1999).

Qiao *et al.* (1999) has tried to classify models in different clusters. These clusters include statistical models, linear programming, network analysis, industrial structure analysis, spatial correlation and location selections. Mathematical models to select locations, include multi-criteria decision analysis methods, and are applied to evaluate decision alternatives based on multiple criteria. These models include the process of selecting evaluation criteria, defining decision alternatives and constraints, criterion weighting and performing decision rules.

Spatial decision support methods are located in the logic tier of the application architecture, communicating with the mobile client and the database. The results of this tier include evaluated decision alternatives, which are sent back to the decision maker.

3.4.1. Modelling Techniques of Spatial and Non-Spatial Data

Researchers developed several models and modelling techniques for DSSs, many of these are of interest or applicable also for spatial decision support. These models are drawn from well-known disciplines such as statistics, operations research or management science, and in most cases there is a connection to spatial data. It is talked about SDSSs if real world problems have a spatial component in one aspect of the decision making process. A spatial component may be pre-processed in an additional step and classified as non-spatial attribute. In advance spatial operations and spatial analysis may be applied for preparing the data set for the modelling process. In this case the model of the SDSS operates on non-spatial or attributive data. For example, spatial analysis determines available and suitable sites for building a new office. In a next step attribute data of the land parcels are evaluated in a financial model. One possible example of the use of modelling is in selecting a facility location. There might be a number of criteria, some of these would be spatial in nature, for such a decision. For instance a school might need to be located near residential areas from which potential pupils would travel. A combustor facility, on the other hand, might need to be located away from populated areas. GIS-based spatial operations could be used to provide an index of suitability for sites for such a facility. The decision maker might have a variety of other factors to weigh up, and techniques such as MCDM to reach a final decision.

3.4.2. Spatial Multi-criteria Decision Analysis for LBS

In information science many researchers recognize that the multi-criteria decision problem is the core of decision support, while others recognize multi-criteria DSS as one of the major classes of DSSs (Rinner, 2005) or consider multi-criteria DSS as a generalization of the DSS concept. These considerations can be done also for SDSSs. Malczewski (1999) suggest that MC-SDSSs emphasise the multi-criteria character of spatial decision making. MCDA is such a fundamental concept, because humans tend to base rational decision on an assessment of multiple decision criteria (Rinner, 2005). Janssen and Rietveld (1990) and Carver (1991) were one of the first analyzing the integration of MCDA with GIS. Methods like, location-allocation algorithms, linear programming, binary programming, inter programming, ideal point analysis, weighted linear combinations and

analytic hierarchy process were integrated or connected with spatial systems or GIS software. Different software products like ArcGIS[10], CommonGIS[11], Idrisi, etc. try to integrate some of these functions using strategies ranging from loose coupling to full integration.

Figure 3.13 presents a schematic workflow for GIS-based multi-attribute decision-making, the simpler type of MCDA procedures (Rinner, 2008). The steps visualized in Figure 3.13 can be summarized following:

- Decision-makers have to agree upon a set of feasible decision alternatives.
- Evaluation criteria have to be selected, upon which a rational selection of an alternative can be based. The criteria have to be metric and standardized in order to make them commensurate.
- The relative importance of each criterion in the criterion set has to be quantified using criterion weights. A decision rule defines the way in which the standardized criterion scores are weighted and combined into an overall evaluation score for each decision alternative.
- From the evaluation scores, a ranking of alternatives can be derived and the top-ranked alternative represents the suggested decision.

The decision process may also involve iterations in which the previously determined input parameters are revised. The feedback and control is an important part of each SDSS by definition.

[10] ESRI ArcGIS is a group of GIS, including ArcReader, ArcEditor, ArcView or ArcInfo. ArcInfo is the most advanced version of ArcGIS, which includes added capabilities for spatial data manipulation, editing, and analysis (http://www.esri.com, accessed June 2008).

[11] CommonGIS was developed by the Frauenhofer Institute and is a tool for decision making in a spatio-temporal context, exploring and analyzing geo-referenced data and visual thinking. CommonGIS is written in Java and can be used also over the web (http://www.commongis.com, accessed June 2008).

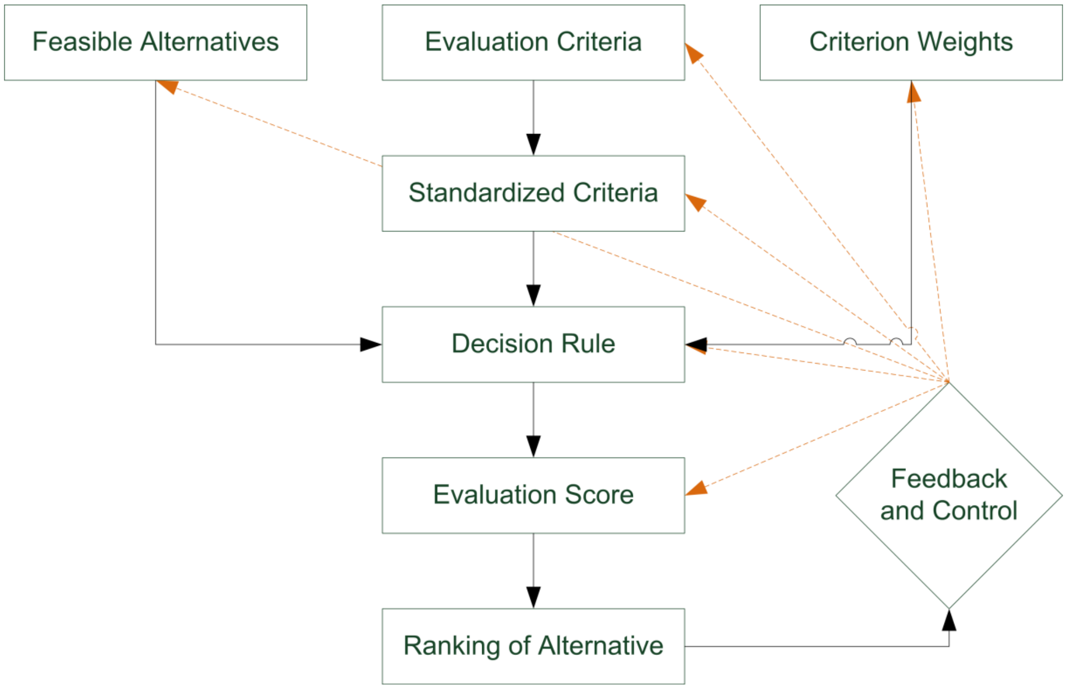

Figure 3.13: Workflow for GIS-based multi-criteria decision analysis, adapted by Rinner (2008).

An integration of MCDA and geo-computation can enhance the GIS-MCDA capabilities of handling larger and more diverse spatial data sets. Benefits of GIS-based MCDA include a formal approach to establishing decision criteria and assessing alternatives. This in term enhances the objectivity and reliability in a rational decision process. Alternatives can be screened efficiently in order to pre-select feasible options. The sensitivity of the outcomes can be analyzed, thus contributing to a feedback phase in an extended decision-making model (Rinner, 2005). On the other hand there is also criticism of MCDA in general. The formal methods are often not practicable for stakeholders, and may result in a misleading decision result to stakeholders' knowledge and intuition. On the technical site the complexity of sometimes large spatial data volumes can limit the practicability of MCDA methods (Heywood *et al.*, 1995).

Spatial multi-criteria analysis is different from conventional MCDM techniques due to inclusion of explicit geographic components. In contrast to conventional MCDM analysis, spatial multi-criteria analysis requires information on criterion values and the geographical locations of alternatives in addition to the common set of evaluation criteria. This means the decision making process include the analysis of evaluation criteria and the geographical distribution of attributes (Ascough *et al.*, 2002). Therefore SDSSs for spatial multi-criteria analysis may use GIS software for the analysis of the geographical component and an MCDM analysis method for further value judgement.

Figure 3.14 presents a three-stage hierarchy of intelligence, design, and choice to represent the decision making process (Simon, 1960) for multiple criteria. In the intelligence phase, data are acquired, processed, and exploratory data analysis is performed. The design phase usually involves formal modelling or GIS interaction in order to develop a solution set of spatial decision alternatives. The GIS functions are critical for supporting the design phase. The choice phase involves selecting a particular alternative from those available. In this phase, specific decision rules are used to evaluate and rank alternatives. The three stages of decision making do not necessarily follow a linear path from intelligence, to design, and to choice (Malczewski, 1999, Ascough *et al.*, 2002).

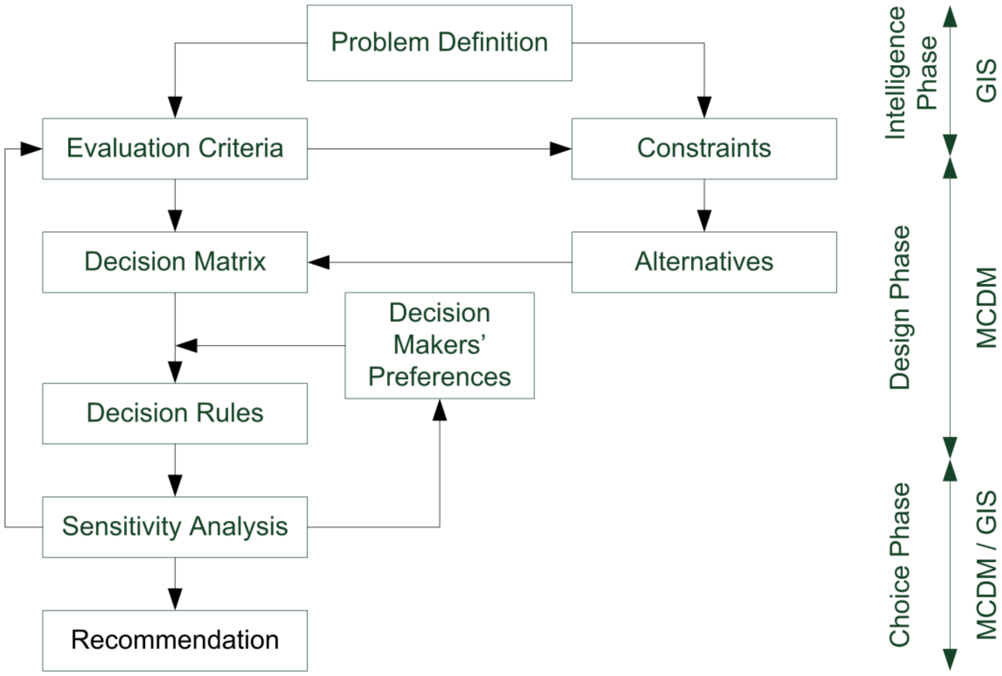

Figure 3.14: Decision flowchart for spatial multi-criteria decision analysis (Malczewski, 1999).

Elements of the spatial multi-criteria decision analysis process have to be adapted for LBDS applications in a way that evaluation criteria, constraints and alternatives are limited and can be formulated as direct input for the decision rule. The decision rule has to be simplified to allow near real time recommendations.

3.4.3. SELECTING EVALUATION CRITERIA

Desirable properties of objectives and attributes can provide guidelines for selection a set of evaluation criteria. Nevertheless, there is no universal rule for determining a set of criteria. It is obvious that the set of criteria necessary for the evaluation is problem specific. That means that evaluation criteria depend on the particular system or decision problem (Malczewski, 1999).

3.4.4. CRITERION WEIGHTING

The purpose of criterion or objective weighting in MCDA is to express the importance of each criterion relative to other criteria. In this step the decision maker can articulate preferences in regard to criteria. A weight is a numeric value assigned to an evaluation criterion that indicates its importance relative to other criteria in the decision situation. The larger the weight the more important a given criterion is. Some of the most popular criterion weighting procedures include:

- raking,
- rating,
- pair-wise comparison, and
- trade-off analysis.

These techniques differ in terms of accuracy, degree of easiness to use and understanding on the part of the decision makers, and in the theoretical foundation (Malczewski, 1999).

The simplest method for assessing the importance of weights is to arrange them in rank order. For this method a straight ranking (most important = 1, second important = 2, etc.) might be used. Based on a rank order there are several methods for generating numerical weights from this information. For example the *Rank sum* weights are calculated according the following formula:

$$w_j = \frac{n - r_j + 1}{\sum(n - r_k + 1)} \tag{3.1}$$

The variable w_j represents the normalized weight for the j^{th} criterion and n is the number of criteria under consideration ($k = 1,2,...,n$), and r_j is the rank position of the criterion. The ranking method is attractive due its simplicity. On the other side the number of criteria limits the practical use of this method.

For rating methods the decision maker estimate weights on the basis of a predetermined scale. In most cases a scale from 0 to 100 is used. A well known rating method is the *point allocation* approach. This method requires the decision maker to allocation 100 points across the criteria of interest, where 0 indicates that the criterion can be ignored and 100 represents the situation where only one criterion need to be considered for the decision analysis. Rating methods are similar to ranking method with similar advantages and disadvantages.

Saaty (1980) developed the pair-wise comparison method in the context of the Analytic Hierarchy Process (AHP). This method involves a pair-wise comparison to create a ratio matrix. The weights are determined by normalizing the eigenvector associated with the maximum eigenvalue of the (reciprocal) ratio matrix (Malczewski, 1999).

The technique consists of taking pairs of criteria C_i, C_j and asking two questions:

- which criterion is more important, C_i or C_j and
- how much is a criterion more important relative to the lesser important criterion.

From these questions a matrix with scores on a 1 to 9 scale (Table 3.2) can be derived and later on normalized.

Table 3.2: Scale of pair-wise comparison (Saaty, 1980).

Intensity of Importance	**Definition**
1	Equal importance
2	Equal to moderate importance
3	Moderate importance
4	Moderate to strong importance
5	Strong importance
6	Strong to very strong importance
7	Very strong importance
8	Very to extremely strong importance
9	Extreme importance

Pair-wise comparison, in contrast to ranking and rating, has a solid theoretical foundation based on ratio-scale judgements about pairs of criteria and the properties of reciprocal matrix of pair-wise comparisons. The disadvantage of pair-wise comparison is a large number of judgements that must be made when the number of criteria is large.

In the process of selecting weights for evaluation criteria it is important to be consistent. Human judgements are proven to be inconsistent for large numbers of evaluation criteria. The rule of transitivity:

$$a > b \wedge b > c \Rightarrow a > c \qquad 3.2$$

is true for pair-wise comparison. If criteria a is more important than criteria b and criteria b is more important the criteria c, then of course criteria a is more important than criteria c. To see the degree of consistency it is recommended to calculate the *consistency ratio* for pair-wise comparison.

3.4.5. DECISION RULES

Decision rules or aggregation methods are procedure that allows for ordering alternatives. The decision rule defines which decision alternative is preferred to another (Malczewski, 1999). For this process data and information on alternatives and decision maker's preferences is integrated into an overall assessment of the alternative. This assessment is expressed often by one score or overall appraisal score. The use of decision rule may facilitate:

- selection of the most desirable alternative,
- sorting of alternatives into classes arranged into a priority order,
- ranking of alternatives from best to worst.

In MADM approach the overall appraisal score is the value of a function that aggregates the outcomes of a decision alternative over all evaluation criteria with the decision maker's preferences. This is why in the MADM context the decision rules are also called aggregation functions. In MODM approach the overall appraisal score is expressed by the value of an objective function measuring the aggregated result of decision variables towards achieving a decision objective. The MADM techniques can be classified according to the level of cognitive processing required from the decision maker and the method of aggregating criterion scores and the decision maker's priorities. According to the cognitive processing level two classes of MCDM techniques can be distinguished:

- compensatory and
- non-compensatory.

The *compensatory approach* is based on the assumption that the high performance of an alternative can compensate for the weak performance of the same alternative on other criteria. In a compensatory MCDM technique the good outcome of an alternative achieved on one criterion is traded off, according to the decision maker's preference structure, for the poor outcome received on another criterion. The term *decision maker's preference structure* means the relative importance assigned to each evaluation criterion. Consequently, a poor outcome of a given alternative on a high-preference criterion may be compensated by a good outcome on another

high preference criterion, or it may require good outcomes on two low-preference criteria to offset a poor outcome on one high-preference criterion. The compensatory approach is cognitively demanding since it requires the decision maker to specify criterion priorities expressed as cardinal weights or priority functions.

Under the *non-compensatory approach* a poor criterion's outcome of an alternative cannot be offset by another criterion's good outcome. In the non-compensatory approach the alternatives are compared along the set of criteria without making intra-criterion tradeoffs. The non-compensatory approach is cognitively less demanding than the compensatory approach since it requires, at the most, the ordinal ranking of criteria based on the decision maker's priorities. For the *conjunctive* decision rule an alternative must exceed a minimum value on all attributes. If an alternative do not fulfil this requirement it is rejected. Accordingly an alternative A_j can be classified as an acceptable alternative only if

$$x_{i,j} \geq x_j^0, j = 1\ AND\ 2\ AND\ 3 \ldots AND\ n \qquad 3.3$$

Where $x_{i,j}$ is the criterion score and x_j^0 is the minimum required level of the j^{th} criterion. Increasing the minimum standard in an iterative way the decision maker can narrow the alternatives down to a single choice.

For a *disjunctive* decision rule an alternative is chosen if and only if it exceeds a minimum required level on one *or* more attributes. Accordingly an alternative A_j is chosen if:

$$x_{i,j} \geq x_j^0, j = 1\ OR\ 2\ OR\ 3 \ldots OR\ n \qquad 3.4$$

Where $x_{i,j}$ again is the criterion score and x_j^0 is the minimum required level of the j^{th} criterion. Sometimes conjunctive and disjunctive decision rules are used in combination. Conjunctive decision rules can be seen as more strict while disjunctive decision rules are more forgiving. Both decision rules can be realized throw simple and composed queries of data.

Following some basic compensatory decision rules are listed as examples for this technique:

SIMPLE ADDITIVE WEIGHTING METHOD

Simple Additive Weighting (SAW) or Weighted Linear Combination (WLC) is one of the most often used techniques for tackling spatial multi-attribute decision making (Malczewski, 1999). They are based on the concept of a weighted average. A total score of each alternative is obtained by multiplying the importance weight assigned for each attribute by the scaled value given to the alternative on that attribute, and summing the products over all attributes. The alternative with the highest score is calculated most suitable according to decision maker's preferences. Expressed with a formula it can be seen that this decision rule evaluates each alternative A_i:

$$A_i = \sum_j w_i x_{i,j} \qquad 3.5$$

where $x_{i,j}$ is the score of the i^{th} alternative with respect to the j^{th} attribute, and the weight w_j is a normalized weight. The weights represent the relative importance of the attribute.

The GIS-based SAW method involves following steps (Malczewski, 1999):

- Define the set of evaluation criteria and the set of feasible alternatives.
- Standardize each criterion map layer.
- Define the criterion weights; that is, a weight of relative importance is directly assigned to each criterion map.
- Construct the weighted standardized map layers; that is, multiply standardized map layers by the corresponding weights.
- Generate the overall score for each alternative using the add overlay operation on the weighted standardized map layers.
- Rank the alternatives according to the overall performance score; the alternative with the highest score or rank is the best alternative.

The SAW method has been implemented in GIS software (e.g., Idrisi) as build-in routines. Basically, the method can be performed in any GIS software with overlay capabilities.

RANK ORDER DECISION RULE

The rank order aggregation method is similar to the SAW method in that the weights are multiplied by alternative outcomes, and then summed, to derive an evaluation score for the option. The same formula as in SAW is used to compute the final appraisal score for each decision alternative. The difference between the two decision rules lies in different procedures of standardizing outcomes of alternatives on evaluation criteria. When using the SAW method, the outcomes are standardized using score range procedure. In rank order technique the outcomes are standardized using *rank order standardisation*. Table 3.3 gives a comparison between score range standardisation and rank order standardisation.

Table 3.3: Difference between score range standardisation and rank order standardisation.

Score range standardisation		**Rank order standardisation**	
Costs	**Score**	**Costs (Rank)**	**Score**
USD 10,000	0.00	USD 10,000 (5)	0.00
USD 1,000	1.00	USD 1,000 (1)	1.00
USD 2,000	0.89	USD 2,000 (2)	0.75
USD 3,000	0.78	USD 3,000 (3)	0.50
USD 3,000	0.78	USD 3,000 (3)	0.50

The interval of change in the standardized criterion score is equal to the change in rank between rank ordered positions divided by the total range in the ranks.

IDEAL POINT METHOD

The *Ideal Point* method orders a set of alternatives on the basis of their separation from the ideal point. The ideal point represents the artificial or hypothetical alternative that consists of the most desirable weighted standardized levels of each criterion across the alternatives under consideration (Malczewski, 1999). The alternative that is closest to the ideal point and at the same

time furthest from the Nadir[12] is considered as the best alternative for this decision rule. Figure 3.15 shows how the *Ideal* point and *Nadir* point can be selected for two beneficial criteria.

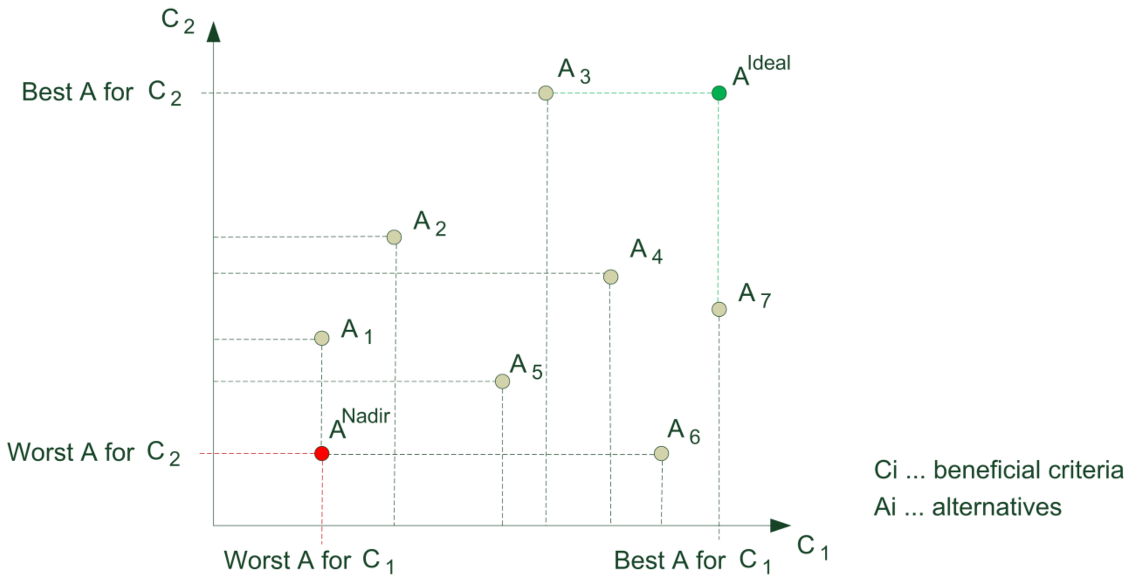

Figure 3.15: Ideal and Nadir point of the Ideal Point decision rule.

The *Ideal Point* decision rule provides complete, interval-scale ranking of decision alternatives. This means that the relative distance of each alternative to the ideal point can be computed. This decision rule avoids the restrictive assumption of independence among the evaluation criteria – made by additive and value/utility function-based decision rules. This makes the *Ideal Point* decision rule an attractive approach to decision problems in which the independence among the criteria is difficult to test. This is especially true in spatial decision problems involving geographically influenced interdependencies or autocorrelations among the evaluation criteria.

CONCORDANCE METHOD

Concordance methods are based on a pair-wise comparison of alternatives. These methods provide only an ordinal ranking of alternatives. When two alternatives are compared, these methods can only express that alternative *A* is preferred to alternative *B*, but cannot indicate by how much. The pair-wise comparison in many methods is based on the extent to which criterion scores and associated weights confirm or contradict the pair-wise dominance relationships between alternatives. There are a wide variety of formulas available to calculate the overall score for each alternative on basis of concordance indicators. Massam (1980) suggest a formula where the appraisal score is the sum of the concordance indices, C_i, for alternative i:

$$C_i = \sum_i c_{i,i'} \qquad 3.6$$

[12] Nadir is a hypothetical alternative consisting of the worst outcomes for the evaluation criteria.

where $c_{i,i'} = \frac{\sum_j w_{j,i,i'}}{\sum_j w_j}$, $\sum_j w_{j,i,i'}$ is the sum of weights for those criteria when alternative i is not worse than the competing alternative i', and $\sum_i w_i$ is the sum of all weights. Based on this decision rule the concordance analysis can be divided in several steps (Malczewski, 1999):

- Determine the set of feasible alternatives.
- Standardise each attribute by transforming the various attribute dimensions ($x_{i,j}$) to non-dimensional attributes ($v_{i,j}$).
- Define the weights (w_i) assigned to each attribute; the set of weights must be such that $0 \leq w_i \leq 1$ and $\sum_i w_i = 1$.
- Generate the concordance matrix by calculating the concordance indices for each pair of alternatives.
- Sum the rows of the concordance matrix to obtain the overall score for each alternative.
- Rank the alternatives according to the descending order of C_i; the alternative with the highest value of C_i is the best alternative.

Further decision rules are *Reasonable Goal Method* (Jankowski *et al.*, 1999), Fuzzy Aggregation Operations like *Fuzzy Additive Weighting* Method or *Ordered Weighted Average* (Malczewski, 1999, Malczewski, 2006).

To deal with inconsistency between evaluation results from different multi-criteria decision rules, decision support toolkits are being developed, which offer access to several methods for comparison of results. Models based on other disciplines such as fuzzy set theory are also adapted to spatial decision-making problems, e.g., the Ordered Weighted Averaging (OWA) method (Jiang and Eastman, 2000, Rinner and Malczewski, 2002, Malczewski *et al.*, 2003, Malczewski, 2006).

4. CONCEPT AND SYSTEM DESIGN

"Design is not just what it looks like and feels like. Design is how it works. [...] Be a yardstick of quality. Some people aren't used to an environment where excellence is expected. [...] Innovation distinguishes between a leader and a follower."

Steve Jobs

The chapter *Concept and System Design* focuses on the idea to combine Location Based Services (LBSs) and Spatial Decision Support Systems (SDSSs) for mobile-based decision support. Technologies from both sides are brought together to design a distributed and service oriented mobile decision support system. In contrast to conventional SDSSs several additional issues have to be considered in the process of design and implementation.

4.1. GENERIC APPROACH TO BUILD AN SDSS

In many cases SDSSs are composed by different types of tools working together for the decision process. For designing and building an SDSS it is useful to examine the different components and their relationships to each other. Densham (1991) discusses the development of DSSs in the context of the framework proposed by Sprague (1980). Sprague's framework allows building a DSS from tools, individual software components, programming languages, programming libraries and small specialized applications that can be combined. For the combination of these components different coupling strategies can be applied (compare section 3.3.3). At a higher level in Sprague's framework are DSS generators, from which a specific DSS can be built (Figure 4.1). Generators may be built from lower level tools. Sprague envisioned that different specific DSS applications would require different combinations of generators and tools. According to this approach specific generators have been designed for certain classes of problems. This framework allows the composition of different combinations to solve different objectives of a decision problem. In other situations general purpose software such as Database Management System (DBMS) packages has been regarded as generators. In DBMSs and GIS software the use of scripting and programming languages facilitates the creation of specific applications. Depending on their key components, DSS generators have strengths and weaknesses. For example, GIS software has strong visualisation capabilities for spatial and attributive data, but only limited support for decision support models (e.g., network analysis models). To create individual models for decision support most GIS software packages provide programming interfaces to extend the functionality of the system.

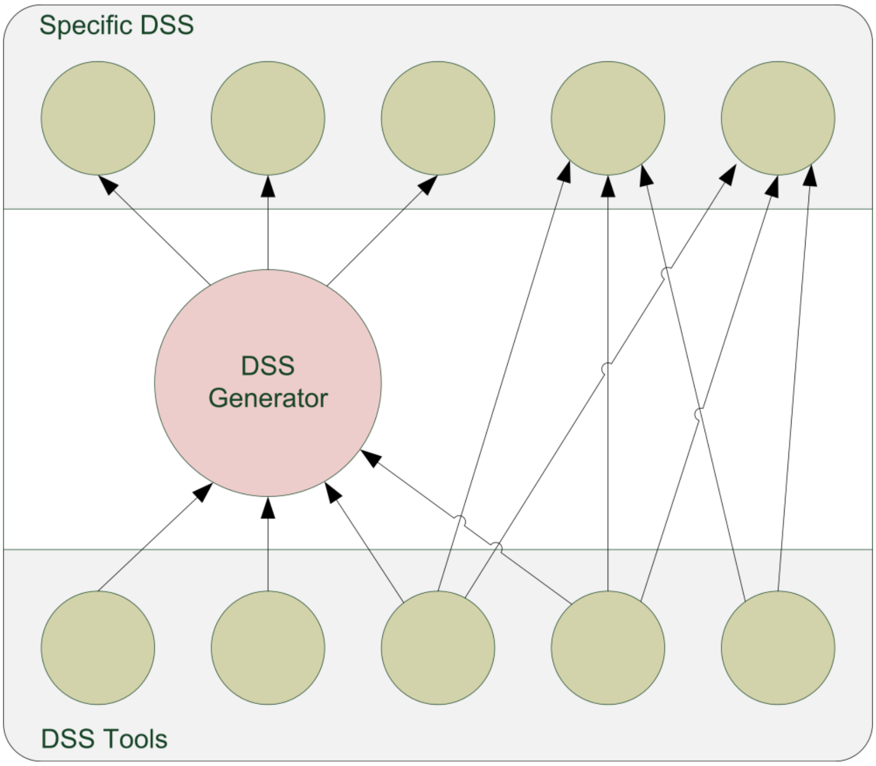

Figure 4.1: DSS generators and the three levels of DSS technology (Sprague, 1980).

The realization of DSS components is strongly dependent on the architecture and the key applications used for decision support.

Table 4.1 provides a comparison of DSS components which can be used for systems with different key applications. For LBSs the UI could be realized with mobile maps or textual description of geographic information or location, data can be queried and composed from distributed web services and geospatial web services and the model can be implemented with programming languages either on the server side or on the client side.

Table 4.1: Realization of DSS components in different key applications.

DSS component	DBMS	GIS	LBS
Interface	tables, forms, reports	multi-layer maps, plots	text, maps
Database	linked database tables	linked spatial and non-spatial databases	spatial and non-spatial web services
Database tools	comprehensive queries	spatial query	querying and composing
Models	basic mathematical functions	basic summarization and network analysis models	basic built-in mathematical functions, server-side optimization techniques
Model building tools	macro and database query languages	macro (script) languages, programming interfaces to other programming languages	APIs, script languages, programming languages

The selection of DSS tools and the composition of them as the use of a generator is an important step in the process of implementing an SDSS. There is evidence that GIS software is becoming increasingly suitable for use as a generator for an SDSS (Qiao *et al.*, 1999). Beside desktop based SDSSs this is also true for server side GISs which can be integrated for LBDS. As GIS designers gain greater awareness of decision making demands, their systems will be designed to facilitate interaction with models. GIS client-side software provides a sophisticated interface for spatial information. Even GIS software with limited functionality, which is available for mobile devices, will provide the ability to perform basic map interaction functions like zoom, pan or identify a feature. GIS provides database support for spatial data and possibilities for spatial data retrieval. However, for the full range of potential uses of spatial data in decision making, a GIS is not a complete DSS because of the almost complete absence of problem specific models or support for the organization of such models. Figure 4.2 focuses on the point that most GIS software lack in the possibility to define an optimization model natively and uses external components which are responsible for decision support methods.

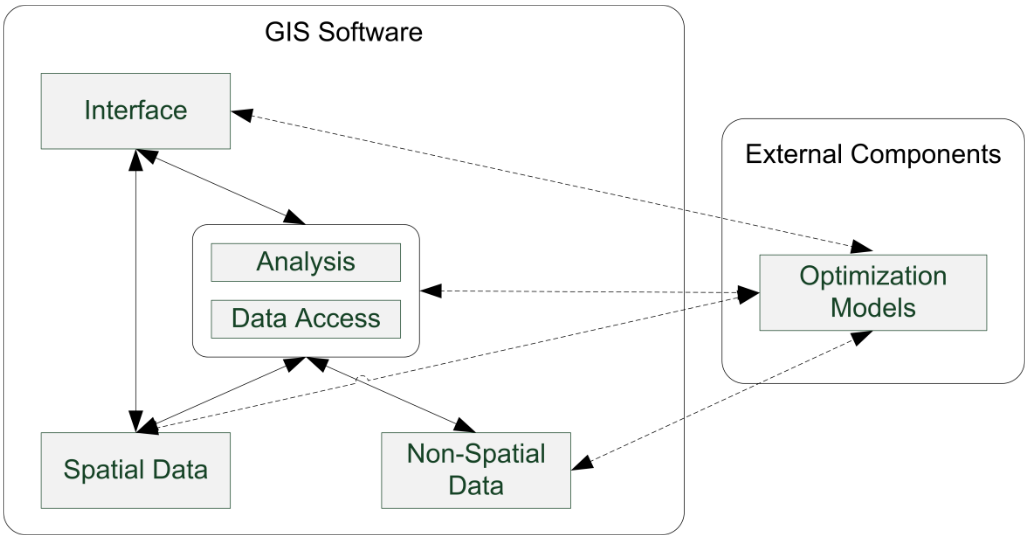

Figure 4.2: Building a SDSS by integrating models with GIS.

Figure 4.2 explains the integration of SDSS models in existing GIS software. It can be explored hat simpler and less effective integration of models only allows modelling routines to access non-spatial data. Spatial data is more complex than non-spatial data and therefore integration of spatial aspects is difficult. Full integration requires that model routines be able to make use of the majority of the features from the GIS. For mSDSSs the design as system with open distributed processing of individual tasks ensures a flexible integration of components.

4.2. System Design for a LBDS Tourist Application

LBDSs are applicable in various different areas, where decisions have to be made immediate at the actual position of interest. In such cases LBDSs running on a mobile device have obvious advantages against traditional SDSSs. Applications for LBDS systems could include tourist guides, emergency management and response, transportation and way-finding, etc. This section describes the system design of a tourist application, which uses web services and multi-criteria evaluation

for a mobile device. The application of a mobile tourist guide supports the tourist in planning activities during a stay in a city. Personalisation of the application is realized through the possibility to weight criteria for the decision support process.

4.2.1. EXAMPLE USE CASE

In contrast to most LBSs for tourism, which represents the user geo-referenced information about tourist attractions and other POIs, the presented application tries to give the user a ranked list of tourist attractions. The ranking is calculated using a MCE, based on user preferences, e.g., personal interests and the actual location. The following scenario shall be solved efficiently with such a system: A tourist visits a city, where she or he is not familiar with. The tourist has huge interest in the culture and architecture of the city and moderate interest in nature and parks as well as shopping and events. She/he wants to see places other people recommend or many people think that these are worth visiting. The tourist wants to have a suggestion of attractive places, and how well they fit to her/his preferences. Based on such a list the tourist wants to make a decision which places are worth to visit for her/him and select these tourist attractions. Additional she/he needs to receive relevant information and the routing to the selected places.

This ordinary example shows the advantage of an mSDSS. The tourist wants to decide which sights are worth to visit for her/him in the current situation. This kind of decisions can be made directly out in the city under consideration of the actual geographic location and personal preferences. For tourist applications recommendations from other visitors are often welcomed. Community generated data, e.g., rating of tourist sights, is very convenient for the integration in the decision support process.

In the following such an application is highlighted from different viewpoints as it is described in the *Reference Model of Open Distributed Processing*.

4.2.2. REFERENCE MODEL OF OPEN DISTRIBUTED PROCESSING

The Reference Model of Open Distributed Processing (RM-ODP) is a set of recommendations and international standards which define essential concepts to specify open distributed processing systems. The RM-ODP is a joint effort by the International Organisation for Standardisation (ISO), International Electrotechnical Commission (IEC) and Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T), with the aim to support distribution, platform and technology independence and portability for Open Distributed Processing (ODP) systems. The focus is on large-scale enterprise software systems to structure the complexity for system specifications but the concepts of different viewpoints can be applied also for smaller distributed systems, like a LBDS.

Important elements of the RM-ODP include:

- an object modelling approach to system specification;
- the specification of a system in terms of separate but interrelated viewpoint specifications;
- the definition of a system infrastructure providing distribution transparencies for system applications; and

- a framework for assessing system conformance.

In nearly every system several people with different interests in the given system are involved, and all these individuals have different reasons for examining the system's specifications. A business executive will ask different questions about the systems than a system implementer. The concept of RM-ODP viewpoints framework, therefore, is to provide separate viewpoints into the specification of a given system. These viewpoints each satisfy an audience with interest in a particular set of aspects of the system. Associated with each viewpoint is a viewpoint language that optimizes the vocabulary and presentation for the audience of that viewpoint.

A viewpoint is a subdivision of the specification of a complete system, established to bring together those particular pieces of information relevant to some particular area of concern, during the design of the system. Although separately specified, the viewpoints are not completely independent. However, the viewpoints are sufficiently independent to simplify reasoning about the complete specification. The mutual consistency among the viewpoints is ensured by the architecture defined by RM-ODP, and the use of a common object model provides the glue that binds them all together. The RM-ODP framework provides five generic and complementary viewpoints on the system and its environment:

- The *enterprise viewpoint,* which focuses on the purpose, scope and policies for the system, specifies the business requirements and how to meet them.
- The *information viewpoint* describes the semantics of the information and the information processing performed. It defines the system information management and the structure and content type of the supporting data.
- The *computational viewpoint* can be considered as the object model or component model which enables distribution through functional decomposition on the system into objects and their interaction as interfaces. It describes the functionality provided by the system and its functional decomposition.
- The *engineering viewpoint* focuses on the mechanisms and functions required to support distributed interactions between objects in the system. It describes the distribution of processing performed by the system to manage the information and provide the functionality.
- The *technology viewpoint* focuses on the choice of technology (hardware, software and conformance) of the system. It describes the technologies chosen to provide the processing, functionality and presentation of information.

Figure 4.3 summarizes all five viewpoints of the RM-ODP and compares them with the software engineering phases: requirement analysis, functional specification, design and implementation.

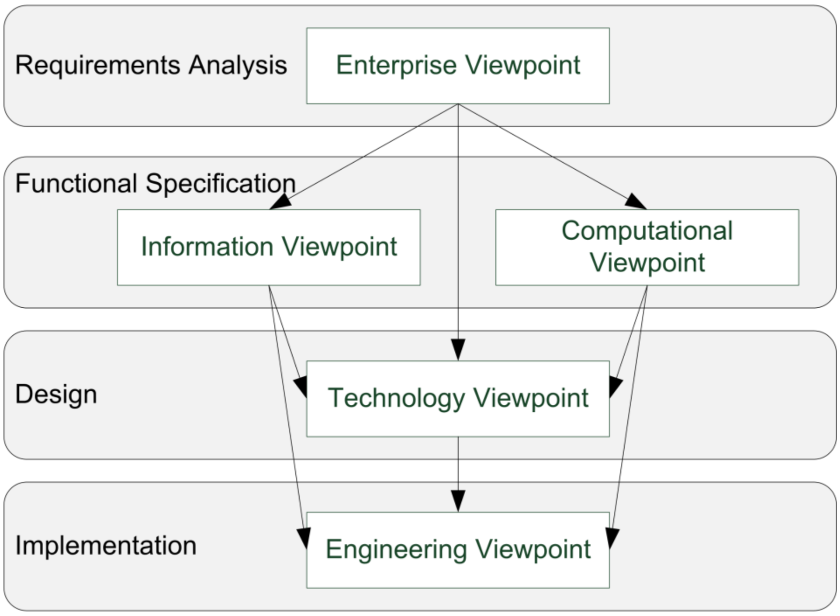

Figure 4.3: Viewpoints defined in the RM-ODP.

The ODP reference model provides abstract languages for the relevant concepts but it does not prescribe particular notations to be used in the individual viewpoints. The viewpoint languages defined in the reference model are abstract languages in the sense that they define what concepts should be used, not how they should be represented. The general purpose modelling notation Unified Modeling Language (UML) is clearly the most promising candidate for a notation, since it is familiar to developers, easy to learn and to use by nontechnical people, offers a close mapping to implementations, and has commercial tool support (Romero and Vallecillo, 2005). However, there is no widely agreed approach to the structuring of specifications of the viewpoint languages of the ODP reference model.

In order to address these issues, ISO/IEC and the ITU-T started a joint project in 2004, called *Information technology - Open distributed processing - Use of UML for ODP system specifications* or simple *UML4ODP*. This document defines use of UML 2, for expressing the specifications of open distributed systems in terms of the viewpoint specifications defined by the RM-ODP. It defines a set of UML profiles, one for each viewpoint language and one to express the correspondences between viewpoints, and an approach for structuring them according to the RM-ODP principles.

4.2.3. ENTERPRISE VIEWPOINT AND REQUIREMENTS

With the enterprise viewpoint tries to specify the scope of the tourist guide application. Figure 4.4 emphasises on the purpose and scope of the system and specifies which actors and use cases are involved in the system process. The diagram tries to show the roles of the actors and which basic activities have to be performed for the objective of the application.

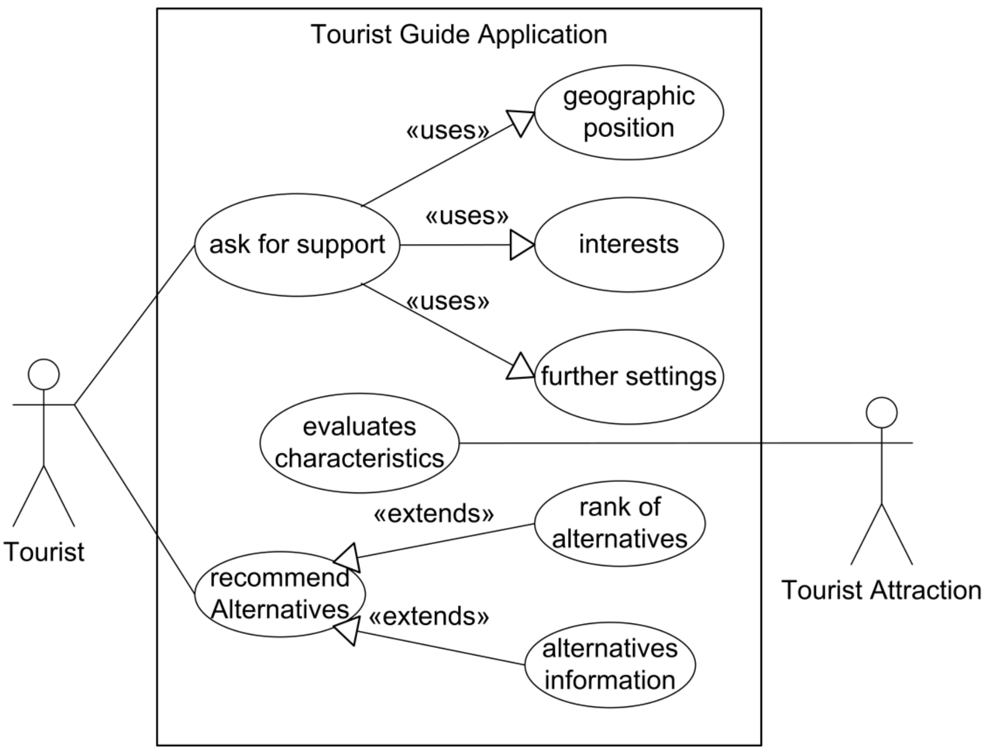

Figure 4.4: Use Case diagram of the tourist guide application.

The *Use Case* diagram of Figure 4.4 identifies actors like *Tourist* and *Tourist Attraction*. The Tourist has the role of the consumer of tourist guide application and wants to visit some tourist attractions within a city. Therefore the user asks the tourist guide for support. The tourist guide evaluates the characteristics of the tourist attractions against the tourist interests, the position and further settings to recommend a list of ranked attractions to the tourist.

Decision alternatives in a spatial scope can be represented using different categories of spatial geometries (compare Table 4.2). For a tourist guide application available tourist sights represent decision alternatives, which can be chosen by the tourist. These tourist places can be represented as a point set including geographical coordinates and attribute information like name, description, etc.

Table 4.2: Representing decision alternatives (Malczewski, 1999).

Geometry	Purpose (Example)
Point	Site selection: points represent a set of potential sites.
Line	Corridor selection: lines represent alternative corridors.
Polygon	Land evaluation problem: Patterns of land parcels represent alternatives.
Point-point	Shortest-path problem: pairs of points represent alternative paths.
Point-line	Bus stop location: lines represent bus routes and points represent potential locations of bus stops.
Point-polygon	School districting problem: points represent schools and polygons represent alternative districts served by the schools.
Line-line	Route intersection problem: pairs of linear objects represent alternatives.
Line-polygon	Flood prevention problem: rivers are represented by lines and potential flooded areas are alternatives represented by polygons.
Polygon-polygon	Hierarchical districting: alternative patterns of nested districts are represented by polygons.
Point-line-polygon	Watershed management: alternative arrangements of point objects (dams) and polygons (reservoirs) with respect to linear objects (rivers);

The information flow and the related activities are represented in Figure 4.5 and include:

- User input of weights for decision criteria via the user interface of the mobile device.
- GPS or other positioning techniques record the actual position of the handheld device.
- Based on the input and actual location a query is formed and needed data is requested from the database.
- The database, including various, distributed data sources delivers relevant data for the model base.
- In a pre-processing step these datasets are prepared for the decision support process using basic GIS functionality like geo-computation or GIS analysis.
- The decision support process itself uses the input criteria and eventually pre-defined parameters to calculate decision alternatives.
- The client receives these decision alternatives and visualizes them together with spatial information about the actual environment of the user.
- Based on the decision alternatives, which are ranked, the user can decide on the problem while moving.

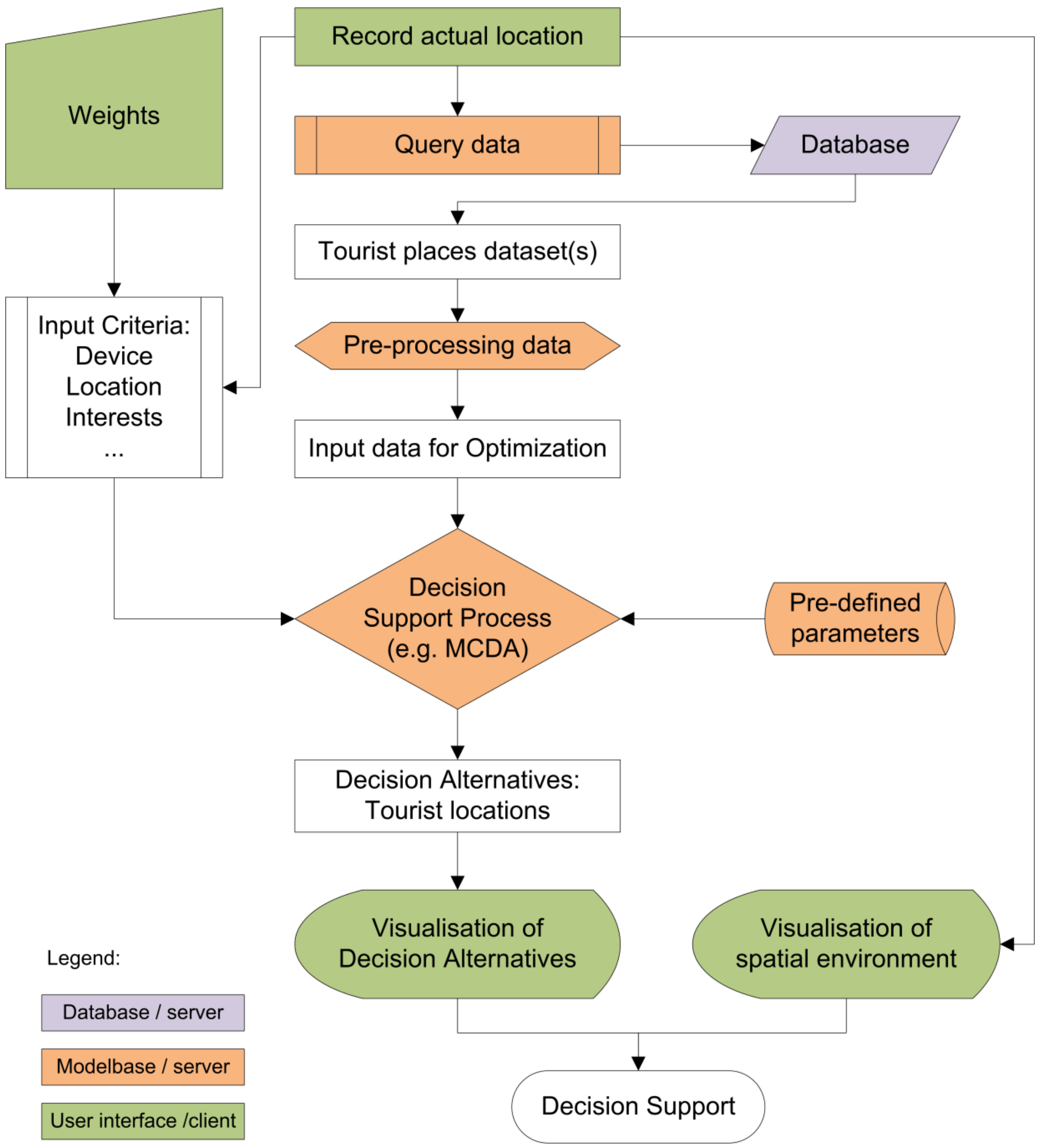

Figure 4.5: Activities for decision support process of a LBDS.

In summary, the client of the tourist guide application transfers a geographical location and the weights for the predefined criteria to the model-base. The model-base generates decision alternatives, namely possible tourist attractions, and their criterion values from the database to perform the MCE. The result is sent back to the client in form of a list including the tourist attractions, their locations and their MCE rank.

4.2.4. INFORMATION MANAGEMENT AND PROCESSING

The information viewpoint is concerned with information modelling. An information specification defines the semantics of information and the semantics of information processing in an ODP system, without considerations about other system details, such as its implementation, or the technology used to implement the system.

Requirements defined before have to be met with information exchange and information processing performed by the system. Figure 4.6 gives an overview about an instantaneous and static view of exchanged information. In the tourist guide application the information view includes criteria. For example, this could be the distance from the current location of the device to the tourist place, interest categories and settings for the decision strategy. Each criterion is connected with a weight, containing the user's emphasis on this criterion. A weight is normally a value ranging between two extreme values, indicating absolute high importance and no importance. Each tourist place is assigned with a set of criteria to evaluate the place according to these criteria and their weights resulting in decision alternatives. The decision alternatives are represented to the user and include the name of the tourist place, a description, the location and of course the rank of the evaluation.

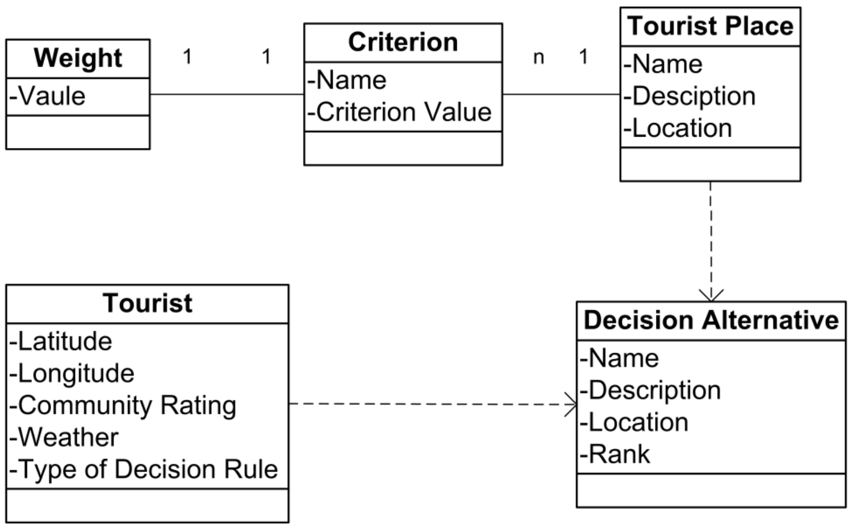

Figure 4.6: Static schema of exchanged information.

The selection of evaluation criteria and data sources for tourist places plays a role to understand the information management and processing.

SELECTION OF EVALUATION CRITERIA

The selection of criteria for MCDM is an important part of the decision process of an SDSS using MCE. Spatial MCDM requires the articulation of the decision objectives as well as the identification of attributes which describes circumstances which have to be considered for archiving the objective. The term evaluation criterion describes both the concepts of objectives and attributes (Malczewski, 1999).

A spatial database is required to build a model based SDSS working with geographical data. The database has to be filled with information that is considered to be relevant to the decision problem. This includes data sets that, in a form, constitute the criteria influencing the decision. For some SDSS a GIS is used for data collection and data preparation.

The selection of appropriate criteria is critical for implementing MCDM models. The selection is strongly dependent on the specific application, the objective, which should be reached with the system and the decision maker itself. For this reason it is hard to find guidelines or formal rules for this design process. One approach to select evaluation criteria is to find a hierarchical structure of attributes leading to the objective of the decision problem. Figure 4.7 illustrates the

relationship between the concepts of attributes and objectives with the example of the tourist guide. The overall objective of the tourist guide application is to suggest the tourist her or his *best* tourist places. To achieve this goal sub-objectives like find the best tourist places, matching with the tourist's interests, minimize distance from the tourist location to the location of the attraction and considering weather information are formulated. These sub-objectives can be achieved when considering attributes like tourist interests, community ratings, tourist location, locations of the tourist attractions and weather conditions. The performance of the first sub-objective is accessed in terms of the attributes *Tourist Interests* and *Community Rating*.

Figure 4.7: Hierarchical structure of the objectives and associated attributes.

The user has the possibility for direct manipulation of the interests and indirect manipulation of her/his location. Other attributes are considered to be state of nature.

The selection of evaluation criteria should be done with respect to the problem situation. If the number of evaluation criteria is defined in such a way that the decision model describes the problem situation as closely as possible, this could lead to an extensive number of evaluation criteria. On the other hand a small number of criteria can be chosen to simplify the decision model. The right way for simplification of the real problem situation is important for the usability of the DSS. Malczewski (1999) describes the importance of comprehensive and measureable attributes as evaluation criteria. He suggests that a set of attributes should be:

- *Complete*: the attributes should cover all aspects of a decision problem;
- *Operational*: They can be used meaningful in the analysis;
- *De-compostable*: they can be broken into parts to simplify the process;
- *Non-redundant*: They avoid problems of double counting and
- *Minimal*: The number of attributes should be kept as small as possible.

An attribute is *comprehensive* if its level for a particular decision problem clearly indicates to what degree the associated objective is achieved. It must be unambiguous and understandable to the decision makers. An attribute is *measureable* if it is practical to assign a number to the attribute for each alternative and to assess the preferences of the decision makers for various levels of the attribute (Malczewski, 1999). Table 4.3 summarizes criteria which are selected for the tourist guide application.

Table 4.3: Criteria used in the tourist guide application.

Criterion	Description
Distance	Describes the distance between the actual location of the device and the location of the tourist place; This criterion is derived from spatial attributes. If the distance is very short the criterion value for distance is very high, because it is easy for the tourist to arrive at the place.
Culture and Architecture	Indicates how well the tourist place represents the category Culture and Architecture;
Shopping and Events	Indicates how well the tourist place represents the category Shopping and Events;
Nature and Parks	Indicates how well the tourist place represents the category Nature and Parks;
Community Rating	Community rating represents the popularity of a tourist place.
Weather	The criterion Weather indicated if the tourist place is dependent on bad weather conditions.

Evaluation criteria are stored in the database, while additional information is coming from third party internet data sources. Different criterion values can be predefined using enumerations. The definition of the values of the criteria has to be done by experts and calibrated for the system. Criterion values are set between 0 and 1 to work on a normalized basis.

4.2.5. COMPUTATIONAL VIEWPOINT

The computational viewpoint describes the functionality of the tourist guide LBDS application and its environment through the decomposition of the system, in distributed transparent terms, into objects which interact at communication interfaces. In the computational viewpoint, application and distributed functions consist of configurations of interacting computational objects. The computational viewpoint is directly concerned with the distribution of processing but not with the interaction mechanisms that enable distribution to occur. The computational specification decomposes the system into objects performing individual functions and interacting at well-defined interfaces.

In Figure 4.8 indicates the decomposition in objects and their communication through interfaces. Basically this is divided into the client application running on the mobile device and the tourist guide model-base. One part of the client application is the location manager. The location manager is responsible to receive the actual geographic position from the GPS receiver or other positioning components. An additional part is the interest manager, which provides an interface where the user can communicate her/his interest in a way that is understandable by the system. The last part of the client application is the settings manager where the user has the possibility to specify further settings related with the decision strategy. The client application manages all three parts and itself has a communication interface with the model base. Furthermore the client is responsible for mapping the current location and tourist places. The model base consists of a data processing unit and the MCE model. The data processing object communicates with the data manager, whose task is the interaction to the database. The MCE model uses the information from the processing unit to perform the decision rule and rank decision alternatives. The ranked decision alternatives are sent back from the data processing unit to the client application.

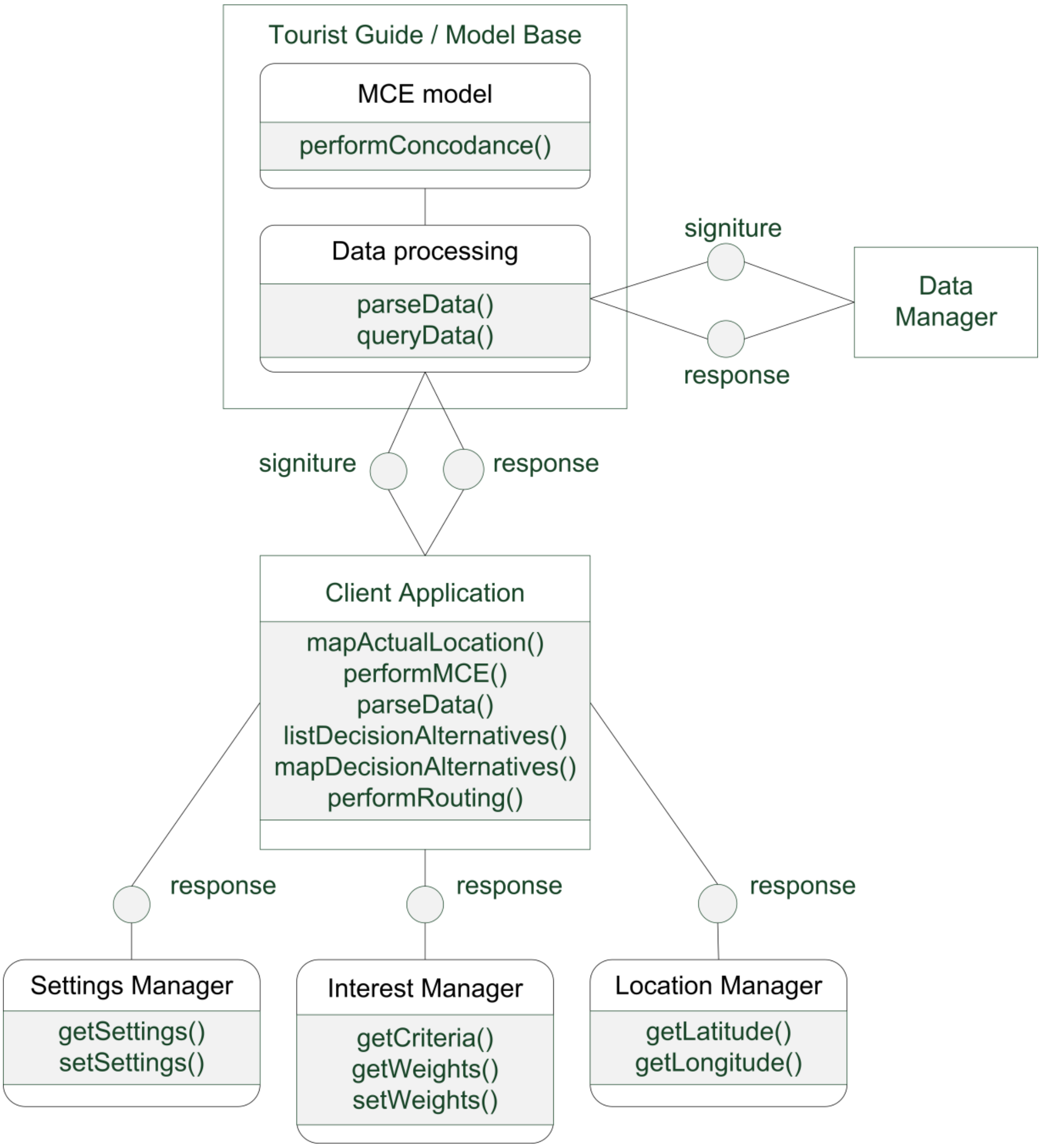

Figure 4.8: Computational viewpoint of the tourist guide application, including functional objects and important operations.

Functional parts of the system and their operations of Figure 4.8 are described in Table 4.4.

Table 4.4: Functional parts of the tourist guide application and their operations.

Functional Part	Operations	Description
Settings Manager	getSettings()	Allows querying all settings from the settings view (e.g., used decision rule, weight for community rating, etc.)
	setSettings()	Sets further settings for the decision process
Interest Manager	getCriteria()	List all Criteria, which influences MCE
	getWeights()	List the weights assigned for criteria
	setWeights()	Set new weights for criteria
Location Manager	getLatitude()	Retrieves the latitude coordinate from the GPS receiver
	getLongitude()	Retrieves the longitude coordinate from the GPS receiver
Client Application	mapActualLocation()	This operation uses coordinates from the Location Manager to visualize the location on a map
	parseData()	Collect all parameters for the MCE and parses them according to the interface definition
	performMCE()	Uses the parsed data to call the MCE method via a web service
	listDecisionAlternatives()	Parses the response from the web service and visualize it to the user.
	mapDecisionAlternatives()	Maps selected decision alternatives according to their rank.
	performRouting()	Calculates a route from the actual position to all selected alternatives.
Data processing	parseData()	Parses the parameters from the client and stores them in variables, which can be accessed by the MCE method.
	queryData()	Queries tourist place data from the database including criterion values.
MCE model	performConcordance()	Calculates decision alternatives and rank them based on the Concordance method.

As example for the operations of the different functional components the *performConcordance* method is described in more detail.

CONCORDANCE METHOD FOR THE PROTOTYPE APPLICATION

The concordance decision rule, described earlier in this work (compare chapter 3.4.5), is applied for the prototype application. The set of feasible alternatives are all available tourist attractions stored in the database or calculated by the model-base. For each tourist attraction the criteria are standardized in a way that the values representing the criteria are between 0 and 1. In this example, there are five criteria distance, community rating, interest for culture and architecture, interest for shopping and events and interest for nature and parks. The standardized criterion value for the distance is calculated using the distance between the current location and the tourist attraction. If the distance is 0 the alternative is most suitable under consideration of this criterion and gets the value 1. If the distance becomes higher the suitability decreases and if an upper level for the distance is exceeded the criterion value becomes 0. The upper level for the distance represents the maximum distance the visitor is willing to walk or travel. The criterion value for the community rating comes from a tourist portal where tourist can rate sights, they already

visited. The values which describe the category or interests are defined by experts. Table 4.5 shows how criteria are arranged for further analysis.

Table 4.5: Example for tourist attractions and their criteria.

Sights	**Criterion 1** (distance)	**Criterion 2** (community rating)	**Criterion 3** (Culture and Architecture)	**Criterion 4** (Shopping and Events)	**Criterion 5** (Nature and Parks)
British Museum	0.3268	0.80	0.95	0.30	0.05
Buckingham P.	0.3373	0.90	0.90	0.00	0.20
Hyde Park	0.0001	0.75	0.20	0.15	0.80
St. James Park	0.4162	0.40	0.10	0.05	0.90
Tower Bridge	0.4935	0.95	0.80	0.10	0.10
Trafalger Square	0.4504	0.70	0.70	0.60	0.00

The weights for each criterion can be selected by using the user interface and are transferred on the server side. With the weights the user has the direct influence for ranking the decision alternatives, while criterion values, instead of distance, are nearly fixed. The weights the method uses for further calculations have to be normalized in the way that the sum of all weights equals one (see Table 4.6).

Table 4.6: Example for tourist attractions and their criteria including weights.

Sights	**Criterion 1** (distance)	**Criterion 2** (community rating)	**Criterion 3** (Culture and Architecture)	**Criterion 4** (Shopping and Events)	**Criterion 5** (Nature and Parks)
British Museum	0.3268	0.80	0.95	0.30	0.05
Buckingham P.	0.3373	0.90	0.90	0.00	0.20
Hyde Park	0.0001	0.75	0.20	0.15	0.80
St. James Park	0.4162	0.40	0.10	0.05	0.90
Tower Bridge	0.4935	0.95	0.80	0.10	0.10
Trafalger Square	0.4504	0.70	0.70	0.60	0.00
Weights	0.25	0.30	0.20	0.10	0.15

With this information the concordance matrix of Table 4.7 can be calculated according the given formula (3.6). As example the technique is shown with three pairs of decision alternatives:

> Concordance set for *Buckingham Palace* and *British Museum* = {criterion 4}, $c_{2,1} = w_4 = 0.10$.
>
> Concordance set for *St. James Park* and *Trafalgar Square* = {criterion 5}, $c_{4,6} = w_5 = 0.15 = 0.15$.
>
> Concordance set for *Trafalgar Square* and *St. James Park* = {criterion 1, criterion 2, criterion 3, criterion 4}, $c_{6,4} = w_1 + w_2 + w_3 + w_4 = 0.25 + 0.30 + 0.20 + 0.10 = 0.85$.

The concordance matrix shows the relative importance between a pair of decision alternatives.

Table 4.7: Concordance matrix.

Sights	British Museum	Buckingham Palace	Hyde Park	St. James Park	Tower Bridge	Trafalger Square
British Museum	-	0.70	0.15	0.40	0.70	0.35
Buckingham P.	0.30	-	0.25	0.50	0.65	0.35
Hyde Park	0.85	0.75	-	0.40	0.75	0.55
St. James Park	0.60	0.50	0.60	-	0.85	0.85
Tower Bridge	0.30	0.35	0.25	0.15	-	0.10
Trafalger Square	0.65	0.65	0.45	0.15	0.90	-
Colum Sum	2.70	2.95	1.70	1.60	3.85	2.20
Rank	3	2	6	5	1	4

According to the given criteria values and weights the result is an ordered list of decision alternatives, where the highest column sum represents the best rank. In this example, the best choice for the tourist will be to visit *Tower Bridge* than *Buckingham Palace* and *British Museum*. The least suitable choice in this case would be *Hyde Park*.

4.2.6. TECHNOLOGICAL AND ENGINEERING PERSPECTIVE

The technological perspective should give the idea which real-world software, hardware and network components are used. This is the starting point for the engineering process. Figure 4.9 shows the technological viewpoint of the LBDS application including their components. The data layer consists of content relevant geographic data, which are in the prototype application information about the tourist attractions to form decision alternatives. Additional data like community data is coming from third-party web services. The logic layer includes elements for data merging and conversion and MCE techniques. In the tourist guide application the concordance method is used as decision rule. The communication between data layer and logic layer is based on the Internet. Also the communication between the mobile client and the logic layer is also done via the Internet using SOAP web services. The client includes a part for the management and communication with the model base, the UI and positioning technology.

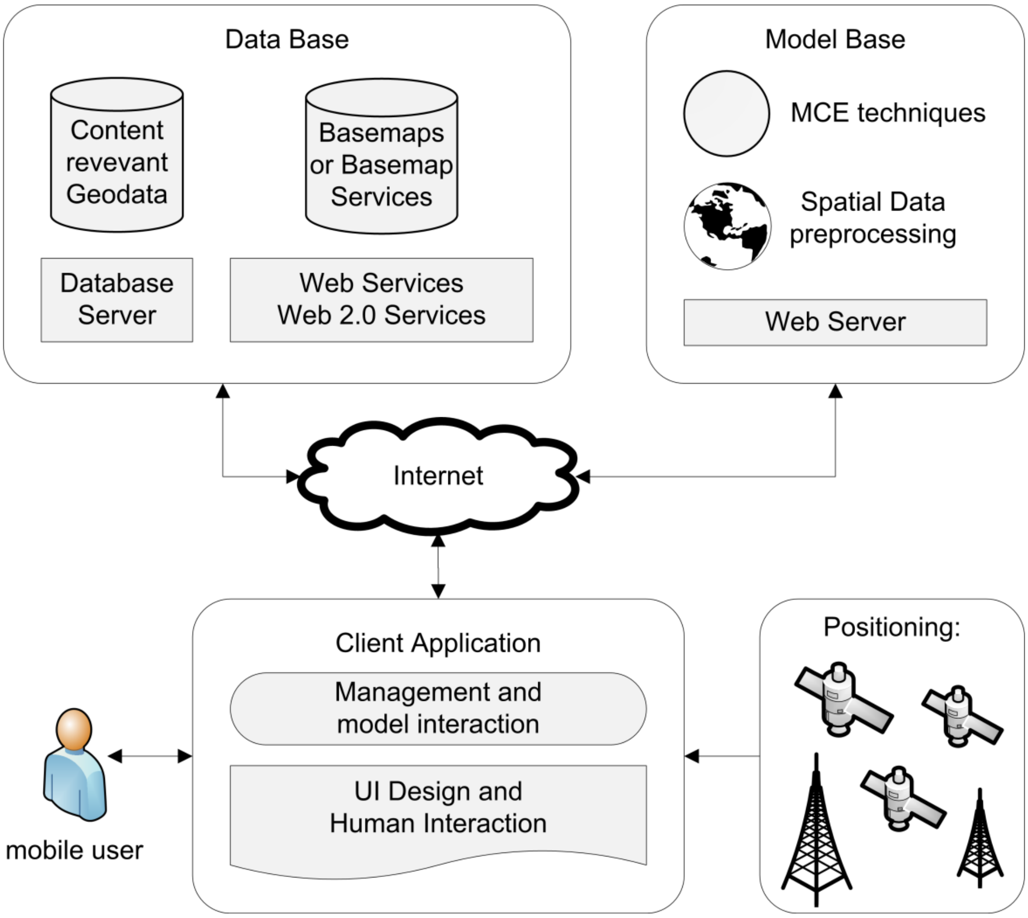

Figure 4.9: LBDS architecture from the technology view.

In order to make full use of the existing resources, speed up the development of SDSS technology, strengthen the depth and scope of its application and increase the benefits of using SDSS, a set of standards concerning data, software and model in the development of SDSS should be established (Yan *et al.*, 1999).

Form the technology view the logic tier holding the model base of the DSS is realized with web services performing MCE. Tasks of the model base include the selection and forming of evaluation criteria, criterion weighting and performing of the decision rule. The next chapter will focus on technological and engineering issues for the tourist guide prototype.

4.3. Summary in Design Strategies

In the chapter, *Concept and System Design*, the development of a distributed framework for a LBDS application is proposed. Several perspectives of the application are emphasised via different viewpoints. The viewpoints focus on a LBDS application as tourist guide but following important assumptions can be derived in general for designing a LBDS application:

- There is a need to define the overall objective of the application and how to suggest an approach to the decision problem. It should also be clarified which sub-objectives have to

be fulfilled to reach the overall goal of the decision problem, and which information is necessary to evaluate the objectives.

- For the design process it is important to specify the activities of the users and the system or system components and which information is exchanged between the actors.
- The decision rule is the mathematical foundation of the decision support process to evaluate decision alternatives. Different decision rules should be evaluated and compared in case of precision and performance for the specific decision problem.
- A further step is to decide which technologies are used for the different components of the system and which communication interfaces are used that these components can work together.

5. Proof of Concept

"In the middle of every difficulty lies opportunity."

Albert Einstein

In this chapter the presented concepts should be realized in a LBS prototype application using MCDA. Concrete technologies and programming languages for the different parts of the applications are described. This LBDS System is designed as application specific DSS and consists of data, model and software for a specific decision problem. The system utilizes different types of Internet technologies and frameworks. Theoretically there is a possibility to share data and models with each other because of the architecture of decision support services. The following subsections describe functionalities and technologies used for the implementation of the tourist guide application.

5.1. Use Case and Functionality

The functionality and interaction of the LBDS system is shown in Figure 5.1 and focused on the objective to provide the tourist places which are worth to visit for her or him. The design study of the user interface explains the functionality and interaction between the tourists and the application. In Figure 5.1a the visitor can specify personal interests in tourist places. Three major categories are provided where the visitor can quantify interests. Each category of interests represents one criterion for the MCE process. In this prototype the criteria are predefined and the user can change the weights of the criterion, which represents the amount of personal interest in this category. The weights of the criteria are normalized and used for the decision support process. By entering settings (Settings Icon, Figure 5.1d) the tourist can enable further criteria like *community rating*, and specify the influence of these criteria in the decision process by changing the weight. With this option the user can set the influence of recommendations from other users for her/his personal final result. Additional weather information can be included and tourist attractions are evaluated due to actual weather situation. If this option is enabled, bad weather conditions can exclude tourist attractions which are depended on weather from the result list. Routing from the current location to the sights can also be enabled or disabled. For more experienced users there is a possibility to choose between different decision-strategies like *Concordance* or *OWA* (Figure 5.1d). All this information is sent to a server, where additional information is collected and pre-processed for the decision support process. After the MCDA process the ranked results are sent back to the mobile client and presented to the user. Figure 5.1b tries to illustrate the results of the decision process. The user gets a ranking of places she/he might be interested in. All sights contain a link to the official homepage, to consume further information about the tourist place. After selecting the final locations, they appear on a map

(Figure 5.1c), where the walking or driving route from the current location of the tourist to the selected sites can be visualised. The map provides basic mapping functionality like *zoom-in* and *zoom-out* or change the style of the base map (Figure 5.1f). At each step of the interaction the user has the possibility to enter a help menu (Figure 5.1e), which gives explanations to the current interaction.

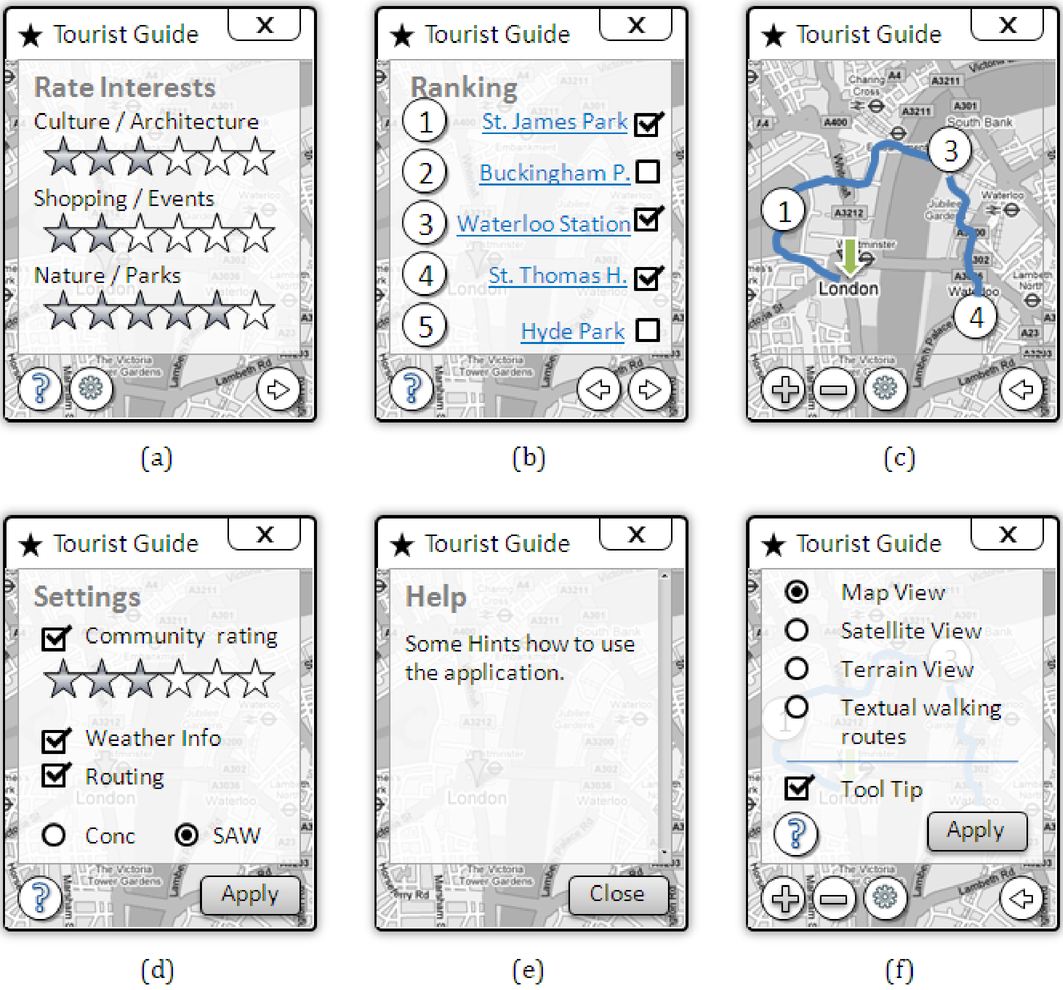

Figure 5.1: UI design of the prototype application.

The design study of Figure 5.1 indicates major steps to operate on a mobile decision support system. These steps are:

- weighting criteria which are used in the decision support process,
- selecting the results from an alternative list, and
- visualization of the results and interaction on the results.

It is important, that at each step the user has the possibility to change her/his criteria to influence the result. The changes in the results affect the alternatives list and the map view. This can be seen as feedback or control step for the decision maker, which has to be included in an SDSS per definition (Malczewski, 1999).

5.2. TECHNOLOGIES TO BUILD A LBDS

Applied technologies can be separated between server side technology and client side technology. For the described tourist guide application the server side includes the data tier and logic tier. The data tier holds data about the tourist places including criterion values. The logic tier communicates with the data tier and the client to perform the decision support analysis. The main task for the client side is the communication with the user and the collection of parameters for the decision support process.

5.2.1. SERVER SIDE TECHNOLOGIES

Server side technologies for an mSDSS include typically an internet server (Apache, Internet Information Server (IIS), etc.) and a map server (ArcGIS Server, UMN-Mapserver, etc.). The web server enables the communication between server and client and transfer of spatial and non-spatial data. The development of applications running on server side are based on server side programming languages like Hypertext Preprocessor (PHP), Active Server Pages (ASP) .NET, Java Server Pages (JSP), Servlets, Python, Perl, etc. The map server software provides a common platform for spatial data and GIS services. Alternative to a map server mapping services like Google Maps, Microsoft Live Search Maps[13] or Yahoo Maps[14] can be used. These map services provide an API for visualizing own spatial data.

To achieve the objectives on the server side of the tourist guide application the logic tier, which includes a MCDA method, PHP[15] was selected as programming language. PHP is a scripting language based on an imperative and object oriented paradigm for producing dynamic web pages. PHP can be embedded into HTML and generally runs on a web server, taking PHP code as an input and creating HMTL web pages as output. It can be deployed on almost any platform and operating system and on most web servers. Since version 5 PHP includes improved support for object-oriented programming as well as other features like the PHP *Data Objects Extension* which is an interface for accessing databases. For PHP the complete source code for users to build, customize and extent for their own use is available free of charge. The *LAMP* architecture has become popular in the *World Wide Web* industry as a strategy of deploying web applications. PHP is commonly used as the *P* in this bundle alongside *Linux*, *Apache* and *MySQL*. In other interpretations of *LAMP* the *P* could also refer to Python or Perl.

To visualize geographic information and maps the *Google Maps* service is used for the tourist application. For this kind of application it is not necessary to set up a map server because it is sophisticated to visualize tourist places as point features. *Google Maps* provides methods to overlay point features, stored in files or in a database, to their base map.

---

13 Live Search Maps (http://maps.live.com, accessed in June 2008) is a mapping service by Microsoft. Live Search Maps combines road and aerial as well as unique bird's eye and 3D view maps for select areas.

14 Yahoo! Maps (http://maps.yahoo.com, accessed in June 2008) is a Yahoo mapping service. The Yahoo Maps APIs allows embedding rich and interactive maps into web and desktop applications using different technologies. The documentation of the Yahoo! Maps APIs is available at http://developer.yahoo.com/maps (accessed in June 2008).

15 PHP stands for Hypertext Preprocessor. The most recent major release of PHP was version 5.2.6 on May 1, 2008.

5.2.2. Client side technologies

For thin client architectures of a decision support system it is easier to implement client software for various client technologies. Basically the client software should run in the browser of the mobile device. This allows certain flexibility on the client side. In the mobile sector a huge variety of different technologies and platforms for implementing software are available. Mobile browsers become more and more useful and can handle similar content as desktop browsers. The client software can be implemented as web interface which is optimized for the screen size of mobile devices. Most actual mobile devices, such as mobile phones, support Java Micro Edition (Java ME). This is an environment for applications running on mobile phones, PDAs and other embedded devices. This platform is portable across many devices and operation systems. However, it can be supposed that in near future mobile devices will support Java Standard Edition (Java SE), because of increasing performance of mobile devices. First hint for this statement is the introduction of *JavaFX mobile* which is a Java operating system for mobile devices such as mobile phones, PDAs, etc. This platform features a Java ME and Java SE implementation running on a Linux kernel. Another platform for mobile devices based on Linux is *Google Android*. This software platform allows writing applications in Java that utilized Google-developed software libraries.

One other possibility to implement applications on mobile clients is Microsoft's .NET Compact Framework (.NET CF). The .NET CF is a version of the .NET Framework and is designed to run on *Windows Mobile* or *Windows CE* based mobile phones and embedded devices. The .NET CF uses libraries designed specifically for mobile devices and some of the class libraries of the full .NET framework. With this technology it is possible to adapt or enhance existing Windows CE applications such as ESRI *ArcPad*.

Important issues for selecting the client technology for the tourist applications are:

- Support of SOA such as web services and SOAP communication.
- Visualisation of geographic content and development of a map interface.
- Easy design and implementation for an intuitive UI.
- Interfaces to implement communication to sensors such as GPS.

5.3. Server Implementation

The implementation on the server side is divided in the data part, which is responsible for the centralized storage of tourist attractions and their criterion values for decision support as well as the application logic in form of a decision strategy and decision rule.

5.3.1. Data Preparation and Management

The database for the prototype application contains information for the tourist attractions and criterion values for each sight. Tourist attractions are represented as point geometry and their location is stored with latitude and longitude coordinates defined by the WGS84 reference system. For the prototype application only the coordinates are stored as numerical values. OGC's Simple Feature Specification (SFS) was not considered in order to keep the implementation simple. Based

on this decision a *MySQL*[16] database was chosen to store potential tourist places. Table 5.1 shows the table holding tourist attractions filled with some exemplary data.

Table 5.1: Database table of tourist places.

ID	Name	Latitude	Longitude	Description
1	British Museum	51.519368	-0.126958	The British Museum is a museum of human [...]
2	Buckingham P.	51.500896	-0.142297	Buckingham Palace is the official London [...]
3	Hyde Park	51.507594	-0.165702	Hyde Park is one of the largest parks in [...]
4	St. James Park	51.502565	-0.134824	St. James's Park is a 23 hectare park in [...]
5	Tower Bridge	51.505663	-0.075155	Tower Bridge is a combined bascule and [...]
6	Trafalgar Square	51.507343	-0.127766	Trafalgar Square is a square in London [...]

The *tourist places* table is connected with criteria, which are essential to perform MCE. There is a one to many connection between the table *tourist place* and the table *criterion*. This means that one tourist place can have one or more criteria and a criterion is connected with exactly one tourist place. Therefore a foreign key is introduced in the *criterion* table to link to a tourist place. One criterion for the MCE is *distance*. This value is not stored in the database, because it is calculated from locations.

Table 5.2: Database table for criteria.

ID	Name	Value	Tourist_ID
1	Culture and Architecture	0.80	1
2	Culture and Architecture	0.90	2
3	Culture and Architecture	0.75	3
4	Culture and Architecture	0.40	4
5	Culture and Architecture	0.95	5
6	Culture and Architecture	0.70	6
7	Shopping and Events	0.95	1
8	Shopping and Events	0.90	2
9	Shopping and Events	0.20	3
10	Shopping and Events	0.10	4
11	Shopping and Events	0.80	5
12	Shopping and Events	0.70	6
13	Nature and Parks	0.30	1
14	Nature and Parks	0.00	2
15	Nature and Parks	0.15	3
16	Nature and Parks	0.05	4
17	Nature and Parks	0.10	5
18	Nature and Parks	0.60	6
19	Community Rating	0.05	1
20	Community Rating	0.20	2
21	Community Rating	0.80	3
22	Community Rating	0.90	4
23	Community Rating	0.10	5
24	Community Rating	0.00	6

[16] MySQL is a cross platform relational DBMS developed by MySQL AB, which is now a subsidiary of Sun Microsystems. The project's source code is available under terms of the GNU General Public License, as well as under a variety of proprietary agreements.

Other important information for the decision support process is coming from a rating platform. The rating platform is a web page where tourist, who has already visited the given tourist places can communicate their impressions by publishing their subjective vote for these tourist attractions. To integrate this kind of community generated information a rating platform was implemented, where the user can set a vote ranging from 1 to 10 for each of the stored locations. In this case 1 means the user does not recommend this sight while a vote of 10 means that the user highly recommends the location for visiting. The actual rating of all tourist sights is coupled with the database table *criterion*.

5.3.2. IMPLEMENTING THE MODEL BASE

The model base or logic tier of the application is realized through a web service. The web service queries information from all tourist attractions, which form the decision alternatives. The service is installed on an Apache web server running on a Linux operating system. As programming language PHP 5 is used to implement the web service and the concordance method is programmed according to the mathematical rules described in chapter 4.2.5. A *user defined object* stores the tourist attractions inside the service to retrieve all necessary information for the calculation.

Since version 5 of PHP a SOAP extension is available, this allows the creation and consummation of web services. The greatest shortcoming of PHP5-SOAP is that there is no way to automatically generate WSDL with it. For this reason the open source project NuSOAP was used. NuSOAP is a group of PHP classes that allow developer to create and consume SOAP web services. It does not require any special PHP extensions and supports much of the SOAP 1.1 specification. However NuSOAP does not provide coverage of SOAP 1.1 and WSDL 1.1 that is as complete as some other implementations, such as .NET or Apache Axis. NuSOAP allows a programmer to specify the WSDL to be generated for the service programmatically.

The following example shows how to create a simple SOAP request and response exchange between client and server.

```
<?php
// Pull in the NuSOAP code
require_once('nusoap.php');
// Create the server instance
$server = new soap_server();
// Initialize WSDL support
$server->configureWSDL('hellowsdl', 'urn:hellowsdl');
// Register the method to expose
$server->register('hello',                  // method name
    array('name' => 'xsd:string'),          // input parameters
    array('return' => 'xsd:string'),        // output parameters
    'urn:hellowsdl',                        // namespace
    'urn:hellowsdl#hello',                  // soapaction
    'rpc',                                  // style
    'encoded',                              // use
    'Says hello to the caller'              // documentation
);
// Define the method as a PHP function
function hello($name) {
        return 'Hello, ' . $name;
}
// Use the request to (try to) invoke the service
$HTTP_RAW_POST_DATA = isset($HTTP_RAW_POST_DATA) ? $HTTP_RAW_POST_DATA : '';
$server->service($HTTP_RAW_POST_DATA);
?>
```

Figure 5.2: NuSOAP example.

The web service is accessed with SOAP protocol to exchange information over HTTP. SOAP is used as protocol because it allows internet communication between programs. SOAP was created to accomplish a HTTP communication between applications running on different operating systems, with different technologies and programming languages.

A SOAP message is an ordinary XML document containing following elements:

- A required *envelope element* that identifies the XML documents as a SOAP message.
- An optional *header element* that contains header information.
- A required *body element* that contains call and response information.
- An optional *fault element* that provides information in case an error occurs while processing the message.

A SOAP message must be encoded using XML using the SOAP envelope namespace and the SOAP encoding namespace to ensure a correct syntax.

Figure 5.3 shows an example of a *concordance* request which is sent to the server. The request will invoke a method called *concordance* and pass a list of parameters including the weights of the evaluation criteria in percent and the current position of the device.

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
   xmlns:soap="http://schemas.xmlsoap.org/soap/envelope">

<soap:Body>

<concordance xmlns="http://www.examplehost.com/touristguide/">
    <w1>70</w1>
    <w2>80</w2>
    <w3>10</w3>
    <w4>30</w4>
    <w5>50</w5>
    <lat>51.479539</lat>
    <lon>-0.100121</lon>
</concordance>

</soap:Body>

</soap:Envelope>
```

Figure 5.3: SOAP envelope to request a web service.

The SOAP response of the *concordance* request is illustrated in Figure 5.4 where a string containing the ordered list of tourist sights defines the result.

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
   xmlns:soap="http://schemas.xmlsoap.org/soap/envelope">

<soap:Body>

<concordance xmlns="http://www.examplehost.com/touristguide/">
    <result>1;St. James Park;51.502565;-0.134824;
            2;Tower Bridge;51.505663;-0.075155;
            3;Buckingham Palace;51.500896;-0.142297;
            4;Hyde Park;51.507594;-0.165702;
            5;British Museum;51.519368;-0.126958</result>
</concordance>

</soap:Body>

</soap:Envelope>
```

Figure 5.4: SOAP response coming from the web service.

WSDL is an XML-based language for describing web services and in particular how to access and locate them. WSDL is a document written in XML and specifies the location of the service and the operations or methods the service exposes. Table 5.3 specifies the major elements of a WSDL document including the *portType*, message, types and bindings.

Table 5.3: Major elements of a WSDL document.

Element	Defines
<portType>	The operations performed by the web service
<message>	The messages used by the web service
<types>	The data types used by the web service
<binding>	The communication protocols used by the web service

An Example of the WSDL document of the *Concordance* web service is shown in Figure 5.5 where all described elements can be identified.

The *portType* element describes a web service, the operation that can be performed, and the messages that are involved. In the example of Figure 5.5 the port type contains an operation called *concordance*. The *message* element defines the data elements of an operation. For the request message seven parameters are defined with data types *integer* respectively *double* and for the response message there is only one element with the data type *string*. The *types* element defines the type that are used by the web service and the *binding* element defines the message format and protocol details for each port.

```
<definitions targetNamespace="http://examplehost.com/soap/SimpleService">
<types>
   <xsd:schema targetNamespace="http://examplehost.com/soap/SimpleService">
       <xsd:import namespace="http://schemas.xmlsoap.org/soap/encoding/"/>
       <xsd:import namespace="http://schemas.xmlsoap.org/wsdl/"/>
   </xsd:schema>
</types>
<message name="concordanceRequest">
   <part name="w1" type="xsd:integer"/>
   <part name="w2" type="xsd:integer"/>
   <part name="w3" type="xsd:integer"/>
   <part name="w4" type="xsd:integer"/>
   <part name="w5" type="xsd:integer"/>
   <part name="lat" type="xsd:double"/>
   <part name="lon" type="xsd:double"/>
</message>
<message name="concordanceResponse">
   <part name="return" type="xsd:string"/>
</message>
<portType name="SimpleServicePortType">
   <operation name="concordance">
      <documentation>Rating of Decision Alternatives</documentation>
      <input message="tns:concordanceRequest"/>
      <output message="tns:concordanceResponse"/>
   </operation>
</portType>
<binding name="SimpleServiceBinding" type="tns:SimpleServicePortType">
<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
<operation name="concordance">
    <soap:operation soapAction="http://examplehostc.com/turistguide/
            concordance.php/concordance" style="rpc"/>
   <input>
      <soap:body use="encoded" namespace="http://www.examplehost.com"
            encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
   </input>
   <output>
      <soap:body use="encoded" namespace="http://www.examplehost.com"
            encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"/>
   </output>
</operation>
</binding>
<service name="SimpleService">
   <port name="SimpleServicePort" binding="tns:SimpleServiceBinding">
      <soap:address location="http://examplehost.com/turistguide/
            concordance.php"/>
   </port>
</service>
</definitions>
```

Figure 5.5: WSDL for the *Concordance* web service.

The example in Figure 5.5 is a request-response operation type, which is the most common, but not the only type which is defined in WSDL.

5.4. Mobile Client Implementation

The implementation on the client was done with the objective to keep client side slim and simple. The user should be provided with a map interface where it possible to interact and navigate. As special function the user can start the MCDA process, specify interest and settings for decision support. An important task of the client is the communication of decision alternatives, and how these alternatives can be used by the tourist.

5.4.1. User Interface Design

User interface design is an important step during the development of an SDSS. The UI should support decision makers through all decision-making phases and is the key to successful use of a decision support system (Ascough *et al.*, 2002). It defines the communication between the user and the system which includes all input and output methods by which data are entered and results and information is displayed. It enables a dynamically interactive session in a real-time exchange of information between the user and the system. Malczewski (1999) listed some issues which should be considered when designing UIs:

- *Accessibility*: This implies that appropriate real-world metaphors are used in developing the graphical environment, and that users unfamiliar with the system can use it intuitively to infer the purpose of a particular screen or graphic object.
- *Flexibility*: This allows the user to recover from unintended and adverse actions.
- *Interactiveness*: This refers to the efficiency of information flow from the user to the system, and vice versa.
- *Ergonomic Layout*: This stresses the effective and efficient communication between the user and the system; several strategies for dealing with the tools contained in the system should be available to the user.
- *Processing-driven*: This allows users to be aware of the tasks they are carrying out; for example, different colours can be used to show active tools or animation in icons to indicate active processing.

A more detailed overview about UI design strategies can be found in Shneiderman (1998). The UI of the tourist application is centred on a map which shows the area surrounding the present location of the tourist. A LBS with map representation requires special cartographic considerations for the visualisation on a mobile device (Reichenbacher, 2003, Müller *et al.*, 2008). As outlined before, the objective for the tourist is to find tourist sights, which fit to entered interests and the current location. The application provides basic map interactions, such as zoom in, zoom out and pan to navigate on the map. A special menu items allows the user to start the MCDA process.

Criteria shown in this view include interests for culture and architecture, shopping and events and nature and parks. The focus of the application is on the map and therefore the weighting of

criteria is kept limited and easy to handle. The approach requires the user to set weights for the predefined criteria with sliders ranging from 1 to 10, where 10 indicates hundred percent interest in the given criteria and 0 no interest.

The result of the MCE is shown on the map. Map icons on the corresponding locations represent tourist attractions. The tourist locations include a number which indicates the rank according the weighting of criteria.

5.4.2. ANDROID PLATFORM

Android is a software stack for mobile devices that includes an operating system, middleware and key applications by the *Open Handset Alliance*.

Figure 5.6 shows the overall architecture of the *Android* platform. *Android* is based on the Linux kernel, but *Android* is not Linux, because it does not include the full set of standard Linux utilities and a native windowing system. The Linux kernel was used for *Android* because it provides a sophisticated memory and process management, security model, driver model and a modularized architecture. On top of the Linux kernel there are native libraries, written in C or C++, providing low level functionalities. An important part of the *Android* architecture is the *Android Runtime* containing the *Dalvik Virtual Machine* and the core libraries. All applications of *Android* are running in a virtual environment, which allows complete independence of application development and the hardware where these applications are running. The core libraries provide the developer with standard utilities, data structures, file access or network access, etc. The application framework is written in *Java* and contains classes and core system services for building applications. In this framework there are core platform services which manage the lifecycle of an application or loading resources. Applications using the application framework form the last part of the architecture.

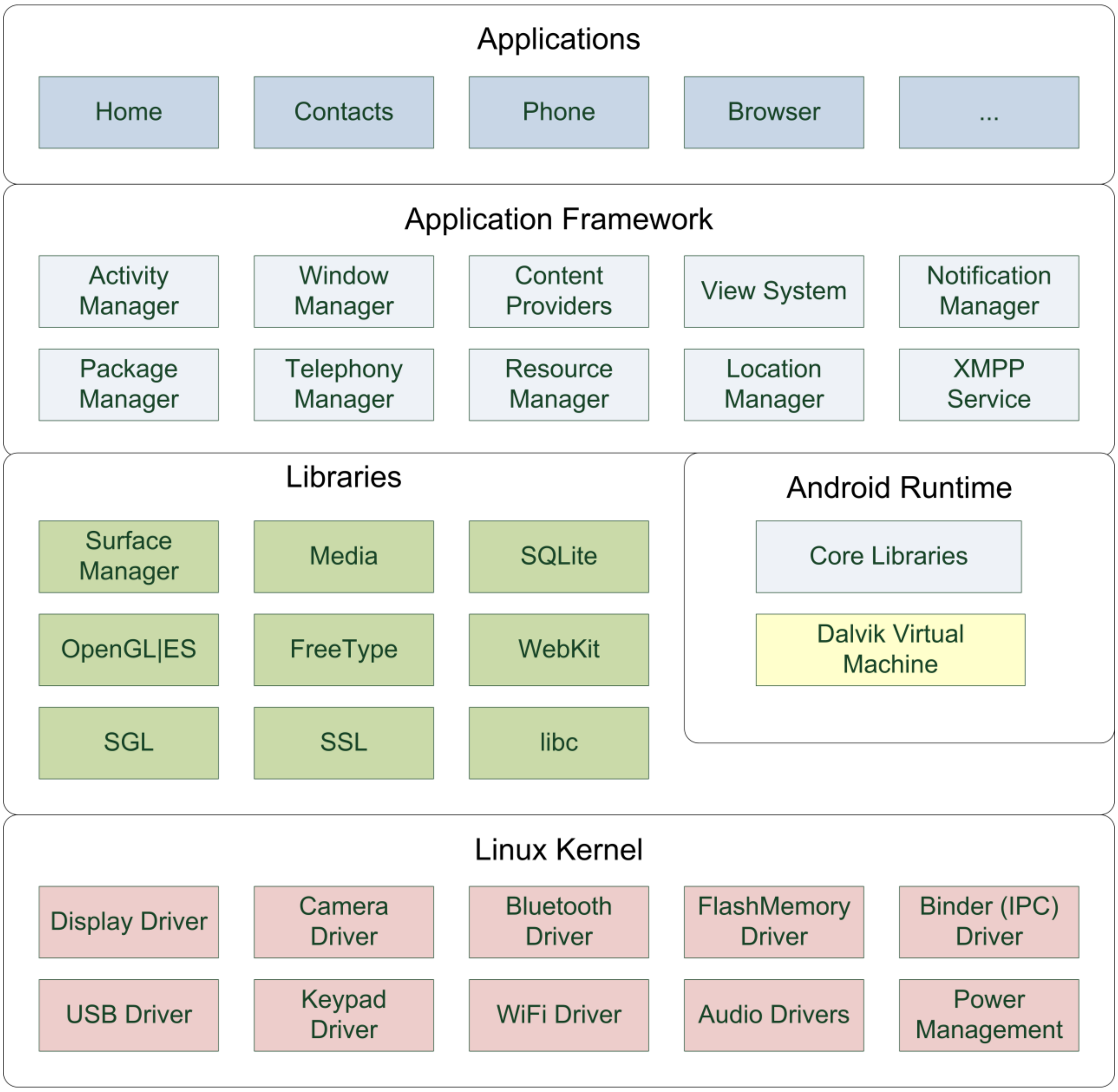

Figure 5.6: Android architecture (Google, Inc.).

The *Android* Software Development Kit (SDK) provides the tools and APIs which are necessary to develop *Android* applications using the *Java* programming language. After installing the SDK several development environments, e.g., Eclipse[17], can be used to implement applications.

There are two ways to create a Graphical User Interface (GUI) in *Google Android*:

- by code
- by XML

[17] Eclipse is a software platform comprised of extensible application frameworks, tools and runtimes for software development and management, primarily written in Java. Its primary use is as an integrated development environment.

XML has the advantage to externalizing the user interface from the rest of the code. Views are building blocks of the UI and could be text views, list views or map views. Views are organized in layouts.

Basic components of an application in *Android* are:

- *Activities:* UI component typically corresponding to one screen.
- *Intent Receivers:* Response to broadcast intents.
- *Services:* Faceless tasks that run in the background.
- *Content Providers:* Enable applications to share data.

There are several different platform in the mobile market which allows to develop own applications. Interesting among them are *Symbian*, *Windows Mobile* and *Google Android*. The Android platform provides application development which is similar to the development of desktop applications. The *Android* SDK includes development and debugging tools, a set of libraries and a device emulator. This makes it easy to start developing and test applications. A further advantage of *Android* are views which represents building blocks of the UI and allow composing application from different types of view, where basic functionality is already included are can be added easily. Integration, extensions and replace are concepts which allow fast development of applications. Android applications are written in *Java* and developers could use any development environment and debugging works the same way as for other *Java* applications.

Figure 5.7 describes the source code of a simple application in Android. Coding is done in Java using special Android packages and libraries. The class *HelloAndroid* extends an *Activity*. The Method *onCreate* is processed when the *Activity* is started. In this method a new *TextView* is created and the text is set to *Hello, Android*.

```
package com.android.hello;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloAndroid extends Activity {
   /** Called when the activity is first created. */
   @Override
   public void onCreate(Bundle icicle) {
       super.onCreate(icicle);
       TextView tv = new TextView(this);
       tv.setText("Hello, Android");
       setContentView(tv);
   }
}
```

Figure 5.7: Java source code of a simple Android application.

5.4.3. IMPLEMENTING AN LBS CLIENT IN ANDROID

To summarize the implementation of an LBS client for Android only some key issues of the application are discussed. In the following *MapView* is described as a basic building block for LBS applications. *MapsViews* are used in combination with the *LocationManager*, which allows

determining the current position of the device. Additionally it is described how to call web services via SOAP from an Android client.

A LBS client is working with location and geographic information in general. For this reason an Android *MapView* is suitable to display geographic information. *MapView* is a building block, which allows displaying a map and performing basic map operations. *MapView* is based on Google Maps using similar data and functionality.

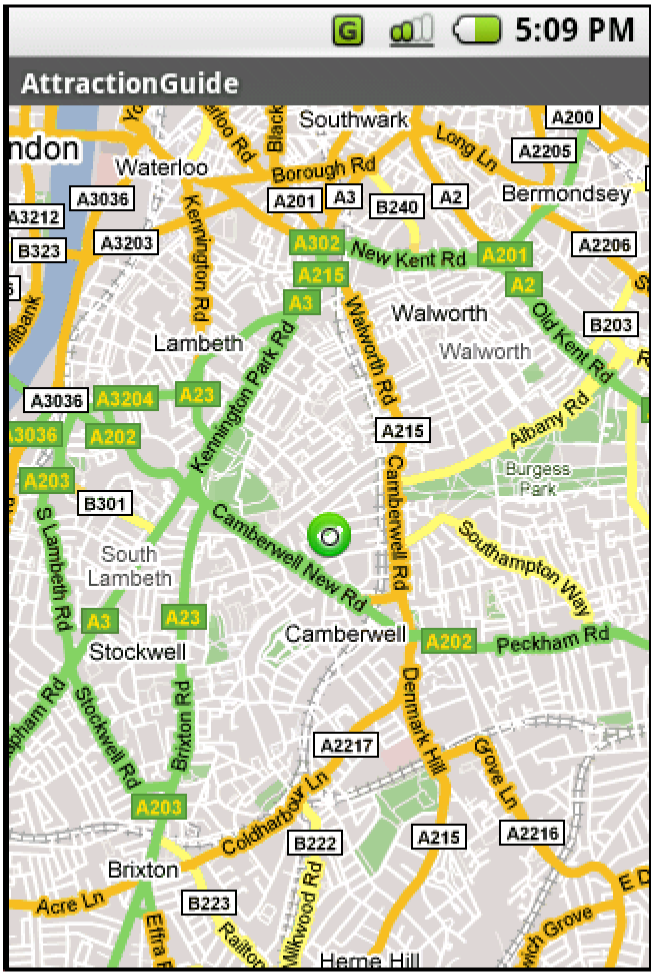

Figure 5.8: Android MapView.

One drawback of the *MapView* in Android is that it is not so easy to create overlays than in Google Maps. Therefore it is necessary to create an overlay class where graphic objects ore pictures can be drawn on the map.

Android provides a *LocationManager* which allows finding the current position of the devices from a GPS receiver. For developing an LBS application it is possible to use a mock GPS provider. Figure 5.9 describes the code to define a *LocationManager*.

```
locationManager =
(LocationManager)getSystemService(Context.LOCATION_SERVICE);

Location location = locationManager.getCurrentLocation("gps");
```

Figure 5.9: LocationManager.

One possibility to access a SOAP Web Service in Android is using an *HttpClient* Object. The code fragment is illustrated in Figure 5.10. It is necessary to specify a *postMethod* with its location, header and body. The header and body represent a SOAP message which includes a SOAP envelope. After the specification of the *postMethod* it can be executed from the *HttpClient* object.

```
HttpClient client = new HttpClient();
String result;
String requestEnv = "<soap:Envelope  ... />; // Put here the full SOAP
request Envelope

PostMethod postMethod = new PostMethod("http://myhost.com/concordance.php");
postMethod.setRequestHeader("Content-Type", "text/xml; charset=utf-8");
postMethod.setRequestHeader("SOAPAction", "http://myhost.com/
concordance.php/concordance");

postMethod.setRequestBody(requestEnv);

int statusCode = client.executeMethod(postMethod);
if ( statusCode == 200 ) {  // good response
       result = postMethod.getResponseBodyAsString();
} else{
       result = "SOAP ERR: "+postMethod.getStatusLine().toString();
}
```

Figure 5.10: Access a web service via SOAP from Android.

If the request works without exception, the result is a SOAP response in XML which can be parsed according to interface definitions.

6. RESULTS

"The simplest way to achieve simplicity is through thoughtful reduction."

John Maeden

This chapter summarizes results from the system design process and implementation of this thesis. The previous section describes the prototypical implementation of the system design proposed in chapter four. The results include the framework to combine spatial decision support methods with LBSs to implement a LBDS system. Additional a tourist guide application is described as LBDS system which includes MCDA to create a personal list of favourite tourist places.

6.1. SYSTEM DESIGN

In the system design process, proposed in this thesis, an open and flexible architecture for LBDS is introduced. The LBDS design is based on an n-tier architecture, which allows the individual and independent implementation and management of each tier. The presentation tier has the task to communicate with the user over a mobile device, while the logic tier is located on the server and includes a MCE method. Data needed to evaluate decision alternatives, are also stored on a server and coupled with additional platforms for adding community generated data.

The design process is done from different viewpoints, each focused on different details of the distributed system. Additional the concordance decision rule is explained in the context of a tourist guide application. Subsequently the tourist guide application is used as show case for a LBDS application.

6.2. PROTOTYPE APPLICATION

The prototype implementation of the tourist guide software is one result of the thesis. Figure 6.1 displays the home screen of the *Android* platform. The interaction with the application is mainly done via the touch screen of a mobile device. The prototype application is presented via several screenshots that describe the LBDS application. By clicking on the software icon the user can start the application. An internet connection displaying map data and querying the web services and a GPS receiver are necessary to use the LBDS client application.

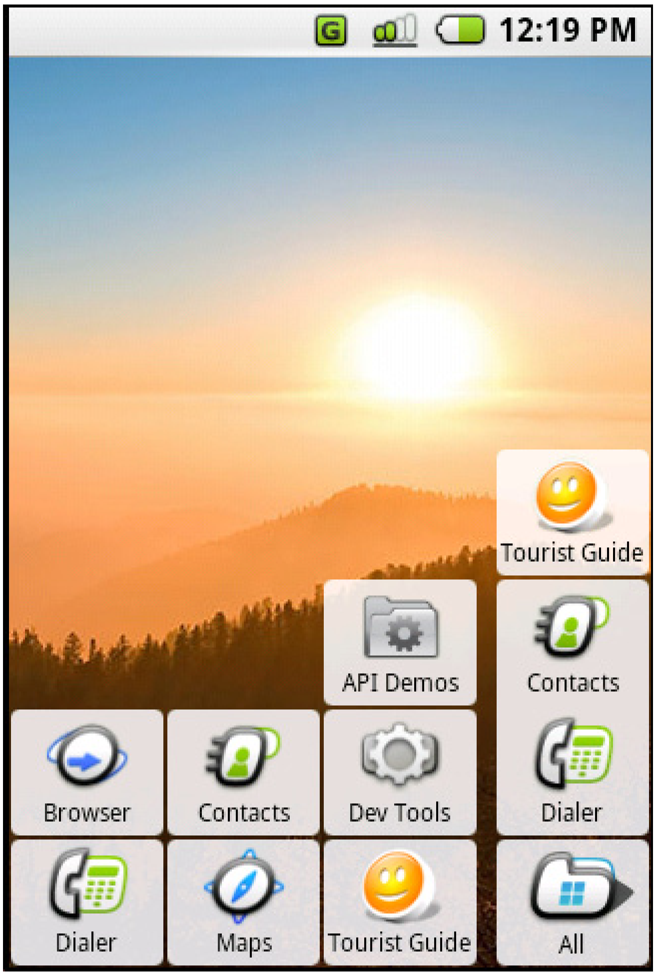

Figure 6.1: Android home screen.

The initial view of the tourist guide application is an Android *MapView*. Figure 6.2 includes a map overlay in form of an icon showing the current position of the tourist from the GPS receiver. The tourist has the possibility to interact on the map and discover the environment. By default the *MapView* is centred to the actual GPS location. In the main menu of the application the user gets simple features like zoom in, zoom out, and change the base map to satellite images or a terrain map (compare Figure 6.3). These features are added as functionality to the main menu, while zooming and panning the map is done over gestures on the touch screen of the device.

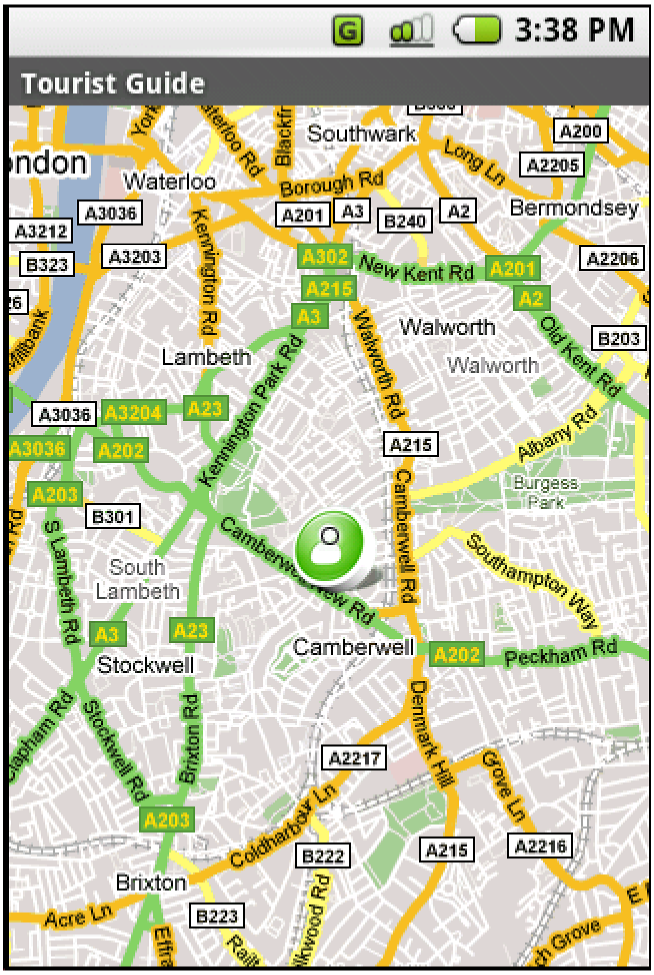

Figure 6.2: The initial view of the tourist guide application.

By calling the main menu the user has the possibility to access the decision support routine. The main menu illustrated in Figure 6.3 includes the menu item *Suggestions* which starts the decision support routine and open the window for the criterion weighting (Figure 6.4). Other functions, implemented in the main menu, includes fixed zoom in and zoom out, change of the base map to satellite view, get direction to perform routing between tourist places and help to access a description how to use the application.

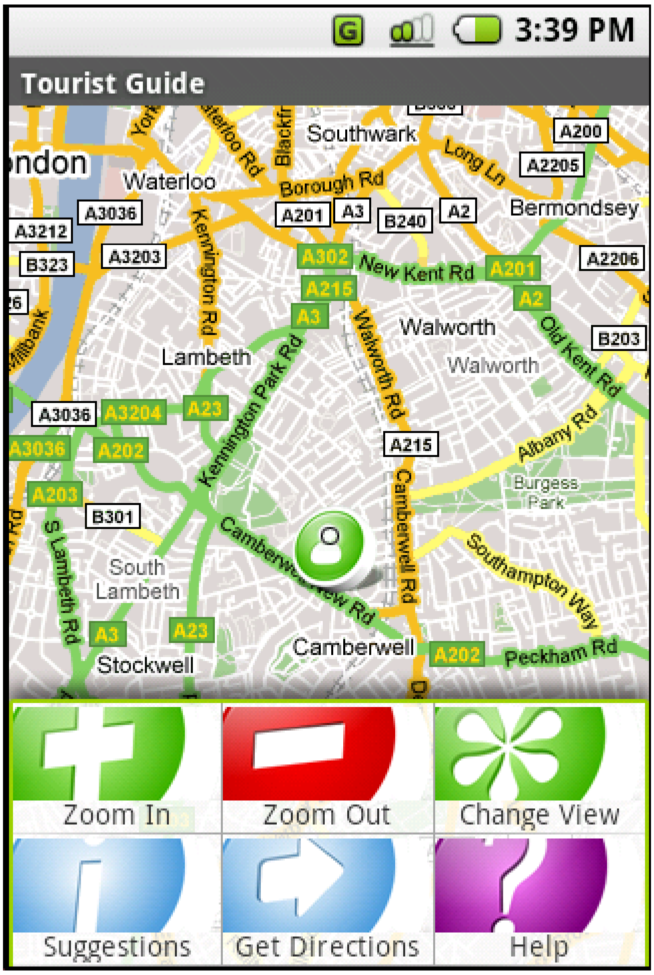

Figure 6.3: Main menu of the tourist guide application.

In the criterion weighting window (Figure 6.4) the tourist can set his or her interest for the predefined criteria *Culture and Architecture*, *Events and Shopping*, and *Nature and Parks*. To make the weighting process simple three sliders are provided by the user interface. The user also gets a feedback about her/his selected weights in percentages. By pressing the *ok* button the weights and other parameters are sent to the service and the MCDA request is started. To specify further criteria the user can also enter the settings view by clicking on the settings button.

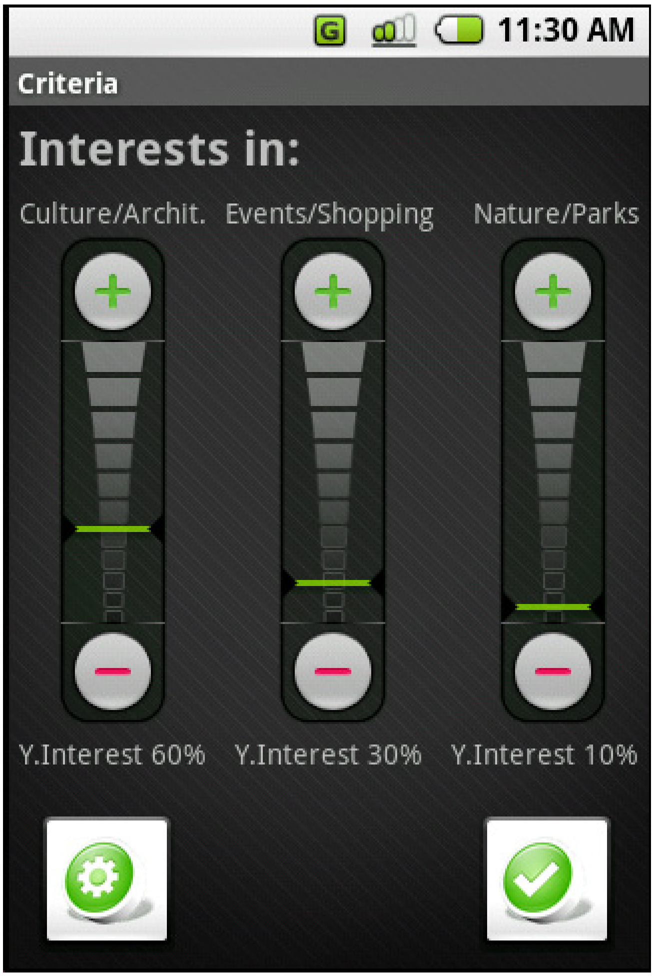

Figure 6.4: Criterion weighting view to specify personal interests.

The settings view, shown in Figure 6.5, holds additional parameters that influence the decision strategy. The user can change these parameters to make further optional customization for the decision evaluation. If there is no change on these parameters default values will be transferred to the decision rule on the server. The tourist can weight the importance of user recommendations for possible tourist places, consider actual weather information and display the routing to tourist locations immediate as result. The option *Include Weather Information* asks for the actual weather conditions at the tourist spots and checks if these conditions recommend a visit to this place. For example, in most cases it is not recommended to visit a park while it is raining. Additional the more experienced user will have the possibility to change between different decision strategies. For the prototype application the weather check and the change of decision strategies are not implemented.

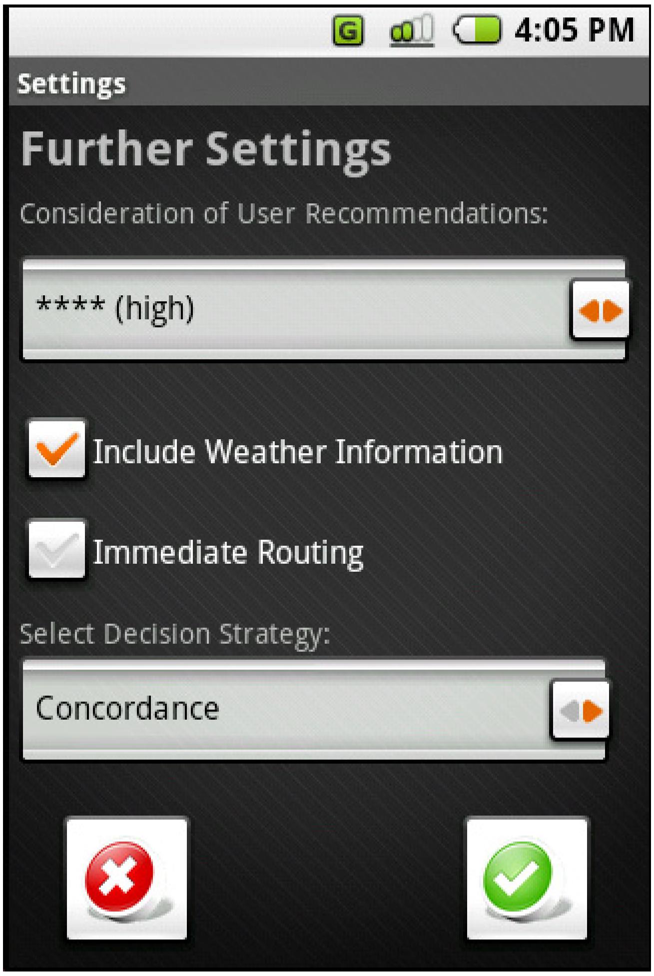

Figure 6.5: Further settings.

After the processing of the MCDA, a result list of alternative tourist places is displayed on the client (Figure 6.6). The user gets the ranked positions of the tourist places and can flag them for visualisation on the map. This view was introduced to give the user the possibility to make a final decision based on the suggested list. Additional information can be retrieved via a button next to one potential alternative.

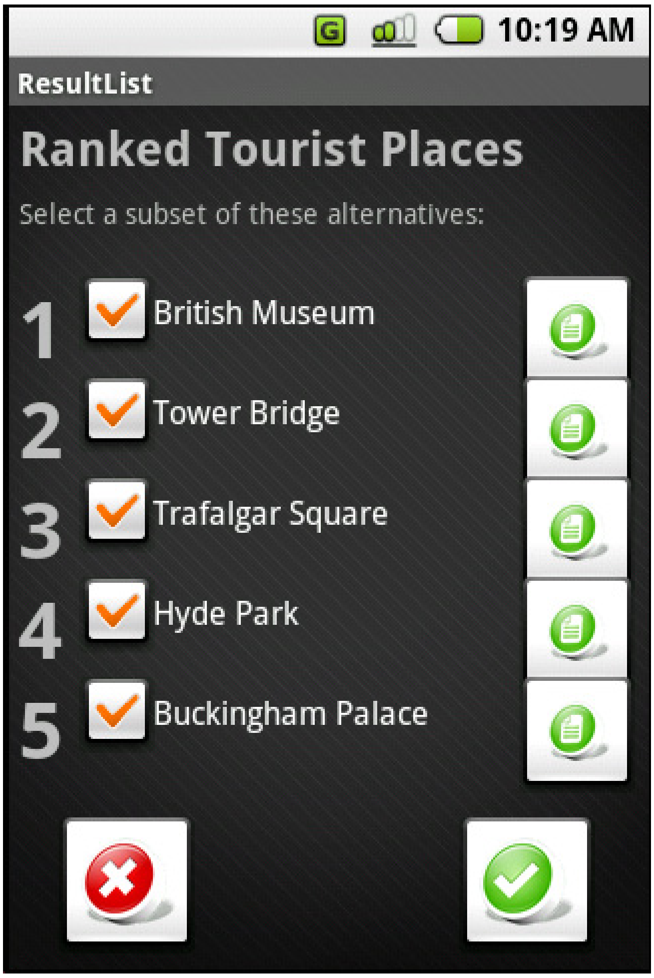

Figure 6.6: Result list of decision alternatives.

Selected alternatives are visualized on the map via an overlay as it is shown in Figure 6.7. The icons indicating tourist places includes information about the rank.

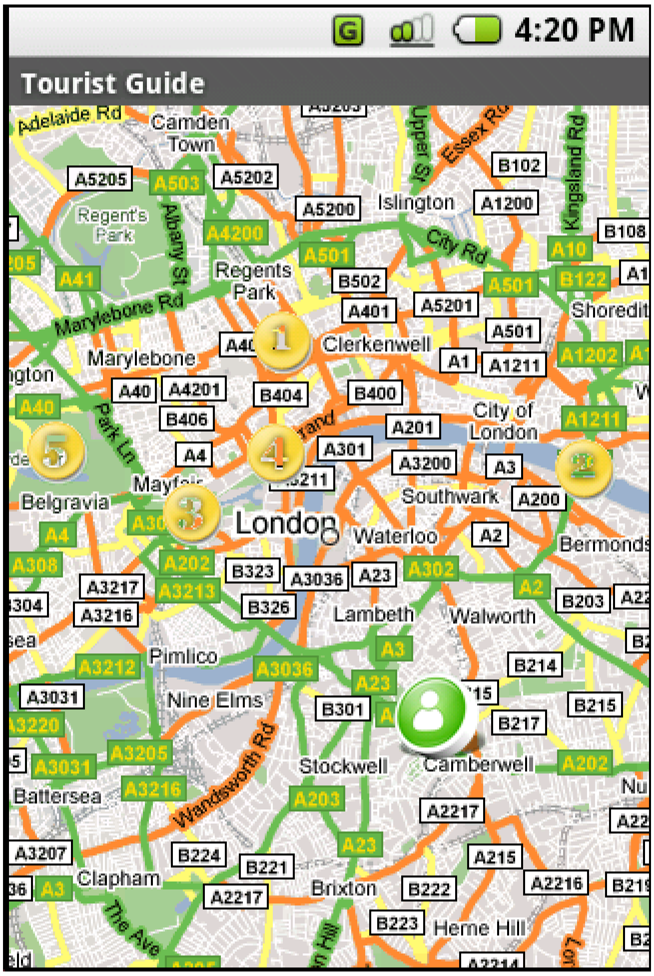

Figure 6.7: Results of the MCDA process.

Depending on the settings the routing from the actual GPS position to the tourist places is shown immediately on the map or the user have to enable the routing over the main menu. Figure 6.8 shows the routing from the user location to all selected tourist attractions.

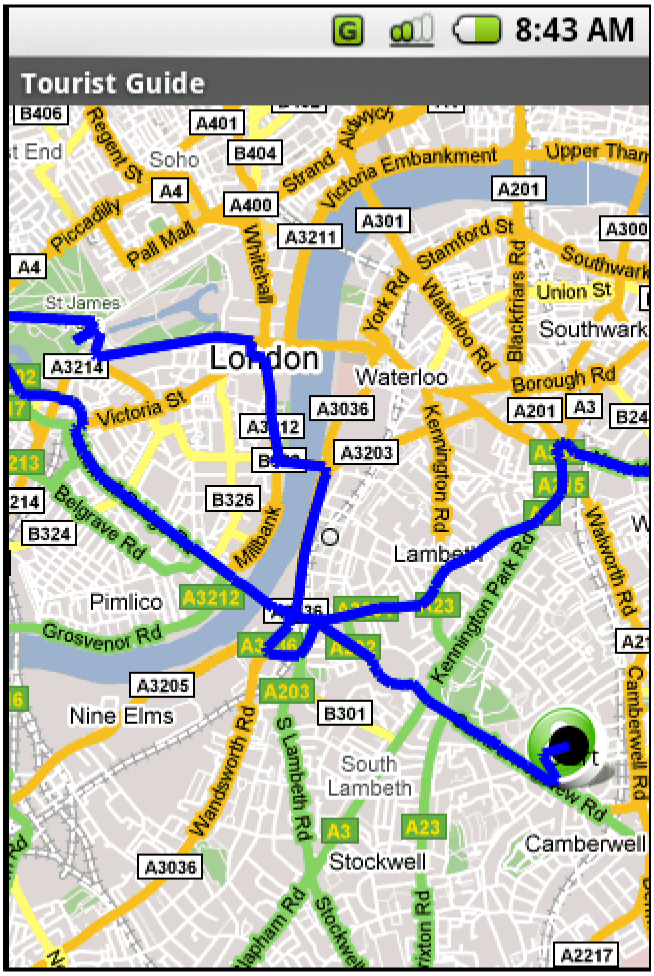

Figure 6.8: Routing to tourist locations.

To include user-generated data for the decision support process a rating platform was implemented. Figure 6.9 represents the screenshot of a web page where tourist have the possibility to rate attractions, they have already visited. This site was implemented to show, how such a rating platform could look like to collect community based data for decision support. The tourist can specify a city she/he has visited and gets a list of attractions. In this case the user can vote from 1 to 10, where 10 indicate a high recommendation for others to visit this location.

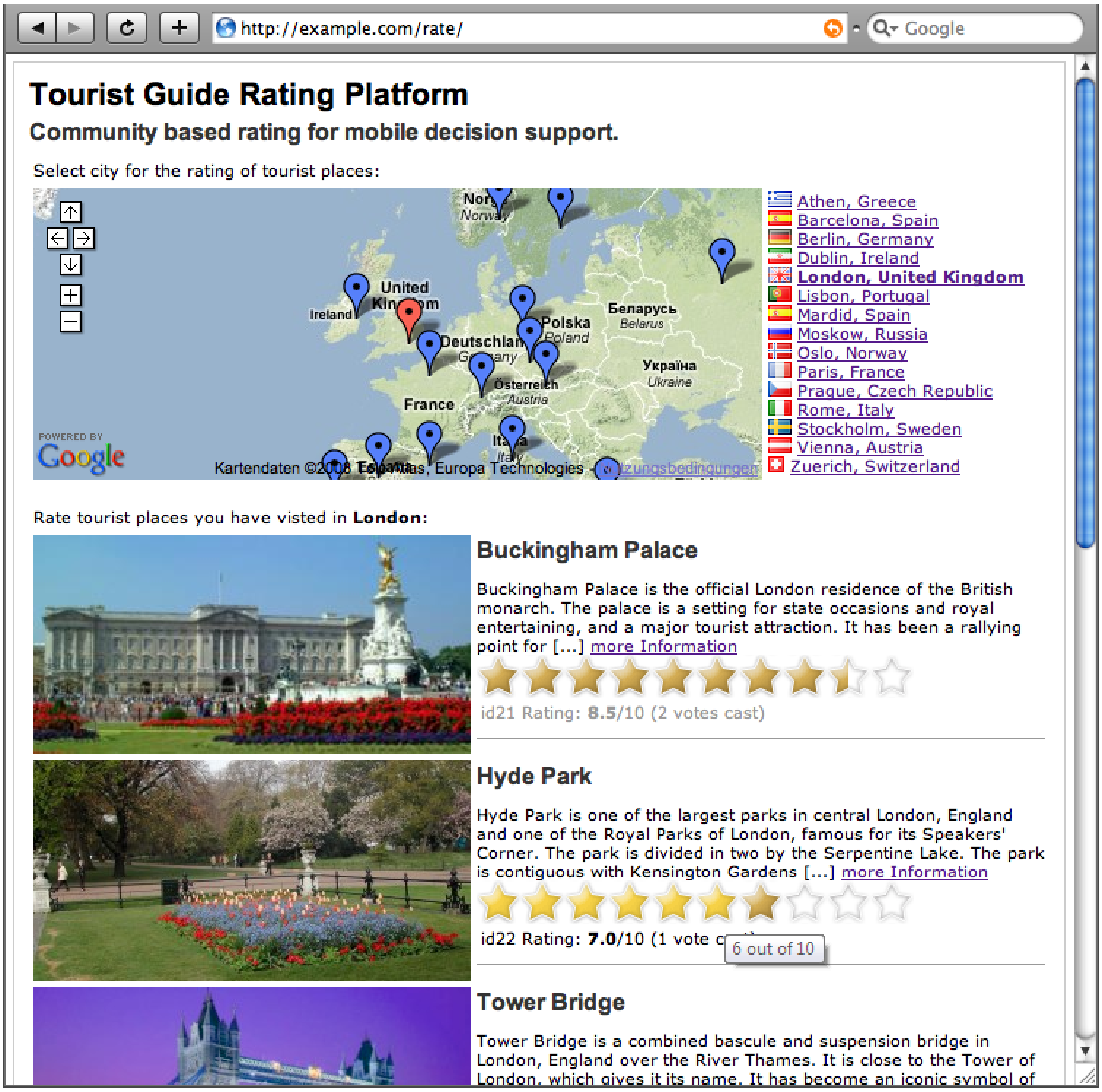

Figure 6.9: Rating platform.

The data is stored in a database and coupled with the database for the decision support process. Instead of this example site, to gain higher page traffic and more votes, third party tourist rating platforms can be used for this purpose. If more votes for a tourist place can be gathered, the significance of the community criteria increases.

6.3. Validation of the Results

SDSSs have the task to assist people in their decision-making and give them information to argue on the taken decision. LBDS systems should support people in their decision-making process during the performance of spatio-temperal tasks. However, until now there are mainly theoretical assumptions how near real-time spatial decision support is working. For this reason it is necessary to test such an application in a field study or human subject test, where user interactions, usefulness and functionality are evaluated with different scenarios. Because of the limited time frame of this work and the lack of resources it was not possible to organize a field test

for the tourist guide prototype application. Instead the application was compared with other LBDS system applied in similar areas. For example, one application is the *Hotel Finder* software (Raubal and Rinner, 2004) where Bäumer *et al.* (2007) describes the results of a human subject test this application. In the human subject test over 70 participants where ask to evaluate each step of the analysis process concerning its intuitive use, accessibility, and readability. The survey was done with questionnaires which were evaluated to get the results of the test. The results of the human subject test are meaningful for other LBDS system and are also considered in the design and implementation of the tourist guide application. The results of the test and expectations of users in a LBDS system can be summarized with following issues:

- Easy user interface in particular for entering parameters for the decision support strategy.
- Additional information to the decision alternatives or selected locations like descriptions links to web pages or photos.
- Routing to locations either walking or driving routes including public transportation.
- Mapping functionality like buffer or measuring tools (e.g., distance in meters and time).
- Wizard guiding through the decision support process.
- Short loading time and immediate results from the decision support methods.

Based on these cognitions, the tourist guide application was designed. For example, it includes a basic routing functionality from the current position to the tourist places and basic information like a description and a link are provided for the decision alternatives. The number of criteria was limited and the way to input criteria weights was optimised for non-experts.

Beside evaluation on the client software also the model including the MCDA has to be investigated in terms of functionality and accurate decision alternatives. It is important to compare different decision rules to each other to detect dependencies on alternatives and differences in the results and the user's decisions as well as the satisfaction of the users with their decisions. Also the criterion values in the databases, which should be set by experts, have to be calibrated and tested in various scenarios to enhance the quality of ranking the alternatives.

7. FURTHER DIRECTIONS AND CONCLUSION

"A decision is as good as the information that goes into it."

John F. Boorout, Jr.

The current section concludes the work and gives an outlook to further directions of the project. It summarizes benefits of an LBS with MCDA discusses design strategies, problems and disadvantages. The outlook focuses on features and considerations which are not discusses in the scope of this work, but are important in developing an LBDS system.

7.1. OUTLOOK

As a next step, it is important to refine the functionality of the prototype application. Missing functions, like the proposed weather check or further decision rules, should be added to get more detailed results. Additional the data set for tourist attractions has to be enlarged to cover a whole city. It is useful to consider further criteria to represent the decision problem in a more realistic way. Opening hours and ticket prices are important for many tourists in finding a decision, either to visit an attraction or not. The implemented rating site can be exchanged with a third party tourist rating platform to use a more balanced community opinion for the decision support process. Based on these improvements a human subject test has to be performed to demonstrate the usability of the LBDS in realistic or near-realistic case studies. This test should cover usability in terms of performance, acceptance, efficiency and user satisfaction. User errors should be recorded, how often and in which phases they appear. One of the most important issue in user interface design is to keep the human-computer interaction short and easy. To shorten the user interaction for weighting criteria predefined profiles, which represents different types of tourist should be provided. If different decision rules are developed it is necessary compare the results of the different methods and calibrate them for realistic use cases. An interesting feature for further consideration is that the map stays visible during the MCDA process, allowing the user to calibrate the MCDA results with his or her perception of the current situation as represented on the map.

From the technological view there are some limitation, due to the fact that *Android*[18] is in an early stage of development. For example, it is very cumbersome to add overlays in the *MapView*. It can be supposed that with future releases of *Android* map overlay will be improved considerably. The routing to tourist locations is currently based on driving directions. In this case it may make sense to use pedestrian routing or public transit and give the user the possibility to choose the routing

[18] For the prototype implementation Android version m5-rc15 (March 3, 2008) was used.

type. Of particular interest for the routing is a combination with open standards like the OGC OpenLS specification.

In general basic mobile decision support could be embedded in standard LBSs, which are used as tourist guides. As additional feature spatial decision support is interesting to many users, for suggesting a planned tourist tour. Different to complex SDSSs, used and operated by experts, it has to be considered that LBDS are mostly used by untrained people. The UI has to be adapted concerning this fact. In future the interaction on LBSs will be done via touch screen, gesture recognition and build in sensors. Mobile decision support for LBSs will be included in many other mobile applications as well.

7.2. SUMMARY

This thesis investigates the combination of spatial decision support techniques and LBSs to create a LBDS. Theoretical issues concerning LBSs and spatial decision support and decision analysis are covered to show the state of the art in the disciplines of decision analysis and mobile location aware applications. The work is influenced by web technologies especially web services and *Web 2.0* concepts. Similar to web-based SDSSs mobile LBDSs are developed for a larger audience of users. Personalisation of services and location aware applications is identified as a main trend in this area. One idea of this work is to integrate personalisation through the implementation of decision support, where the user has the possibility to judge on different criteria. The personal judgement of the criteria influences the result displayed on the map interface of the mobile device. This is related to the first hypothesis of the work:

> *Personalisation of mobile geoinformation services can be done with the integration of spatial decision support methods.*

With the possibility to rate different criteria the user includes a personal opinion in the process of selecting decision alternatives. The advantage of this kind of personalisation is that users need not create and maintain profiles including personal information, which are stored on remote systems. Of course security and privacy issues have to be considered if the person can be identified.

A key contribution is the suggestion of an open framework for implementing a LBDS. The framework is based on an n-tier architecture for distributed systems, where each tier is independent from other parts of the application, connected with well defined communication interfaces. The design is describes though different viewpoints. Community generated content was integrated in the system to reduce the complexity of the decision process. This idea refers to the second hypothesis:

> *Geoweb services and geoweb 2.0 services can simplify the decision process and reduce the complexity of the overall decision strategy in a location-based decision service.*

In form of rating information user-generated data was integrated to in a decision process to select suitable locations. Subjective perceptions of people influence the order of supposed decision alternatives. The integration of user-generated information for decision support can be described as indirect form of collaborative decision making.

Recent developments in the mobile industry predict increasing use of LBSs because of accomplishments in the usability of mobile applications. These developments can be seen as secondary or enabling technologies for LBSs. The design and implementation of the prototype software shows an example among different application fields for LBDSs. The tourist guide software serves as exemplarily scenario where spatial decision support is applicable for mobile real time problem solving. Continued changes in computer hardware and software technology are fundamentally altering the way that decision makers and people in general derive information for their decisions. The trend for SDSSs will probably bring more focus to the needs of the decision makers, and hence the diversity of different systems will rise. In general, SDSSs should move decision making toward an equilibrium which reflects a sophisticated level of technology and an acceptable standard of uncertainty, risk and usability.

"Multicriteria spatial decision support is clearly the appropriate paradigm for the future, precisely because it is such an adaptable and comprehensive concept." (Ascough *et al.*, 2002)

8. LITERATURE

Adams, P.M., Ashwell, G.W.B. & Baxter, R., 2003. Location-Based Services — An Overview of the Standards. *BT Technology Journal,* 21 (1), 34-43.

Ahn, Y.S., Park, S.-Y., Yoo, S.B. & Bae, H.-Y., 2004. Extension of Geography Markup Language (GML) for Mobile and Location-Based Applications. *Computational Science and Its Applications – ICCSA 2004.* 1079-1088.

Armstrong, M.P. & Densham, P.J., 1990. Database organization strategies for spatial decision support systems. *International Journal of Geographical Information Systems,* 4 (1), 3-20.

Armstrong, M.P., Densham, P.J. & Rushton, G., 1986. Architecture for a microcomputer based spatial decision support system. *Second International Symposium on Spatial Data Handling.* International Geographics Union, 120-131.

Armstrong, M.P. & Ruggles, A.J., 2005. Geographic Information Technologies and Personal Privacy. *Cartographica: The International Journal for Geographic Information and Geovisualization,* 40 (4), 63-73.

Ascough, J.A., Rector, H.D., Hoag, D.L., Mcmaster, G.S., Vandenberg, B.C., Shaffer, M.J., Weltz, M.A. & Ahjua, L.R., 2002. Multicriteria Spatial Decision Support Systems: Overview, Applications, and Future Research Directions *iEMS 2002 Integrated Assessment and Decision Support*, Lugano Switzerland, 175-180.

Barnes, S.J., 2003. Developments in the M-commerce value chain: Adding value with location-based services. *Geography,* 88 (4), 277-288.

Bäumer, B., Panov, I. & Raubal, M., 2007. Decision Improvement through Multi-Criteria Strategies in Mobile Location-Based Services. *In* A. Car, G. Griesebner & J. Strobl (eds.) *Geospatial Crossroads GI Forum: Proceedings of the First Geoinformatics Forum.* Salzburg: Wichmann.

Beaubrun, R., Moulin, B. & Jabeur, N., 2007. An Architecture for Delivering Location-Based Services. *IJCSNS International Journal of Computer Science and Network Security,* 7 (7), 160-166.

Beinat, E., 2001. Location-based Services - Market and Business Drivers. *GeoInformatics* 4(3), 6-9.

Benson, J., 2001. LBS technology delivers information where and when its needed. *Business Geographics,* 9 (2), 20-22.

Booth, D., Haas, H., Mccabe, F., Newcomer, E., Champion, M., Ferris, C. & Orchard, D., 2004. *Web Services Architecture, W3C Technical Reports and Publications*.

Botts, M., Percivall, G., Reed, C. & Davidson, J., 2007. OGC Sensor Web Enablement: Overview and High Level Architecture. *OGC*.

Brimicombe, A.J., 2002. GIS - Where are the frontiers now? *GIS 2002*, Bahrain, 33-45.

Cai, G., Bolelli, L., Maceachren, A.M., Sharma, R., Fuhrmann, S. & Mcneese, M., 2004. GeoCollaborative Crisis Management: Using Maps to Mediate EOC–Mobile Team Collaboration, Demo Abstract *5th Annual NSF Digital Government Conference*, Los Angeles, CA.

Caruso, C., Colorni, A. & Paruccini, M., 1994. A decision support system for solid waste management. *In* A.Q. Curzio, L. Properetti & R. Zoboli (eds.) *The management of municipal solid waste in Europe: economic, technological and environmental perspectives.* Amsterdam: Elsevier, 99-112.

Carver, S., 1991. Integrating multicriteria evaluation with GIS. *International Journal of Geographical Information Science,* 5 (3), 321-339.

Casey, M.J. & Austin, M.A., 2002. Semantic Web Methodologies for Spatial Decision Support. *DSIAge2002: International Conference for Decision Support Systems in the Internet Age.* Cork, Ireland.

Castle, C.J.E. & Longley, P.A., 2005. A GIS-Based Spatial Decision Support System for Emergency Services: London's King's Cross St. Pancras Underground Station *Geo-information for Disaster Management.* Berlin, Heidelberg: Springer

Crossland, M., D. & Wynne, B., E., 1994. Measuring and testing the effectiveness of a spatial decision support system. *In* J. Nunamaker, .F. & R. Sprague, H (eds.) *Twenty-Seventh Hawaii International Conference on System Sciences.* Hawaii IEEE Comput. Soc. Press.

Cutter, S.L., 2003. GI Science, Disasters, and Emergency Management. *Transactions in GIS,* 7 (4), 439-445.

Densham, P.J., 1991. Spatial decision support systems. *In* D.J. Maguire, M.F. Goodchild & D.W. Rhind (eds.) *Geographical information systems: principles and applications.* London: Longman, 403-412.

Diamond, J.T. & Wright, J.R., 1988. Design of an integrated spatial information system for multiobjective land-use planning. *Environment and Planning,* 15 (2), 205-214.

Eastman, J.R., Jin, W., Kyem, P.A.K. & Toledano, J., 1995. Raster procedures for multi-criteria/multi-objective decision. *Photogrammetric Engineering and Remote Sensing,* 61 (5), 539-547.

Elson, J., Howell, J. & Douceur, J.R., 2007. MapCruncher: Integrating the Worlds Geographic Information. *Operating Systems Review,* 41 (2), 50-59.

Eom, S., Lee, S. & Kim, J., 1993. The intellectual structure of Decision Support Systems (1971-1989). *Decision Support Systems,* 1993 (10), 19-35.

Erden, T. & Coskun, M.Z., 2007. Interfacing Emergency Management witth GIS-Aided Spatial Decision Support Systems. *International Symposium on Modern Technologies, Education and Professional Practice in Geodesy and related Fields.* Sofia, Bulgaria, 270-279.

Ewart, G.M., 1994. Integrating geographic information systems and multi-criteria decision making to develop a prototype decision support system for health professionals. University of Idaho.

Faber, B., Wallance, W. & Cuthbertson, J., 1995. Advantages in collaborative GIS for land resource negotiation *9th Annual Symposium on Geographic Information Systems*, Vancouver, BC: GIS World, Inc., 183-189.

Fielding, R.T., 2000. Architectural Styles and the Design of Network-based Software Architectures. University of California.

Fischer, G., Markowski, M. & Antione, J., 1996. Multiple criteria land user analysis. *Working Paper WP-96-006.* Laxenburg, Austria: International Institute for Applied Systems Analysis.

Fonseca, F.T. & Egenhofer, M.J., 1999. Ontology-Driven Geographic Information Systems. *In* C.B. Medeiros (ed.) *7th ACM Symposium on Advances in Geographic Information Systems.* Kansas, MO, 14-19.

Ganter, J.H. & Smith, J.D., 1995. MPATHav: a software prototype for multiobjective routing in transportation risk assessment *11th international Conference on the Packaging and Ransportation of Radioactive Materials*, Las Vegas, 992-999.

Gonzales, M., 2003. The new GIS Landscape. *Intelligent Enterprise,* 2 (2003), 21-24.

Goodchild, M.F., 2000. The current status of GIS and spatial analysis. *Journal of Geographical Systems,* 2 (1), 5-10.

Goodchild, M.F., 2008. Assertion and authority: the science of user-generated geographic content *Proceedings of the Colloquium for Andrew U. Frank's 60th Birthday. GeoInfo 39*, Department of Geoinformation and Cartography, Vienna University of Technology.

Guarino, N. & Giaretta, P., 1995. Ontologies and Knowledge Bases: Towards a Terminological Clarification. *In* N.J.I. Mars (ed.) *Towards Very Large Knowledge Bases.* IOS Press.

Heywood, I., Oliver, J. & Tomlinson, S., 1995. Building an exploratory multi-criteria modelling environment for spatial decision support. *In* P. Fisher (ed.) *Innovations in GIS 2.* London: Taylor & Francis, 127-136.

Hilton, B.N., 2007. *Emerging Spatial Information Systems and Applications* London: Idea Group Inc (IGI).

Hjelm, J., 2002. *Creating Location Services for the Wireless Web* New York: John Wiley.

Huerta, E., Navarrete, C. & Ryan, T., 2005. GIS and Decision Making: A Literature Review. *In* J.B. Pick (ed.) *GIS in Business.* PA: Idea Group Publishing.

Jankowski, P., 1995. Integrating geographical information systems and multiple criteria decision-making methods. *International Journal of Geographical Information Science,* 9 (3), 251 - 273

Jankowski, P. & Eward, G.M., 1996. Spatial Decision Support System for health practitioners: selecting a location for rural health practice. *Geographical Systems,* 3 (2), 279-299.

Jankowski, P., Lotov, A. & Gusev, D., 1999. Application of a Multiple Criteria Trade-off Approach to Spatial Decision Making. *In* J.C. Thill (ed.) *Spatial Multicriteria Decision Making and Analysis.* Aldershot: Ashgate, 127-148.

Jankowski, P. & Nyerges, T.L., 2001. *Geographic Information Systems for Group Decision Making* London: Taylor & Francis.

Janssen, R. & Rietveld, P., 1990. Multicriteria analysis and geographical information systems: an application to agricultural land use in the Nethernalds. *In* H.J. Scholten & J.C.H. Stillwell (eds.) *Geographical information systems for urban and regional planning.* Dordrecht, The Netherlands: Kluwer Academic, 129-139.

Jarupanthirum, S. & Zahedi, F.M., 2005. GIS and spatial support systems. *In* J.B. Pick (ed.) *GIS in Business.* PA: Idea Group Publishing, 151-174.

Jiang, H. & Eastman, J.R., 2000. Application of fuzzy measures in multi-criteria evaluation in GIS. *International Journal of Geographical Information Science,* 14 (2), 173-184.

Jun, C., 2000. Design of an Intelligent Geographic Information System for Multi-criteria Site Analysis. *URISA Journal,* 12 (3), 5-17.

Keenan, P., 1995. Spatial Decision Support Systems for Vehicle Routing. *Working Paper MIS 95/10.* University College Dublin, Graduate School of Business.

Keenan, P., 2007. Using a GIS as a DSS generator. *In* R.K.J. Bandamutha & P. Saxena (eds.) *Geographic Information System.* Hyderabad: ICFAI University Press, pp. 97-113.

Kolodziej, K.W. & Hjelm, J., 2006. *Local Positioning Systems: LBS Applications and Services* Boca Raton: Taylor & Francis Group.

Kwan, M.-P. & Schmitz, B.C., 2002. Privacy Protection and Accuracy of Spatial Information: How Effective are Geographical Masks? *GIScience 2002 (Second International Conference on Geographic Information Science).* Boulder, Colorado.

Lee, G.-S., 2005. Fuzzy Multi-criteria Decision Making-Based Mobile Tracking. *In* O. Gervasi (ed.) *International Conference of Computational Science and its Applications, ICCSA.* Singapore: Springer Verlag, 839-847.

Longely, P.A., Goodchild, M.F., Maguire, D.J. & Rhind, D.W., 2005. *Geographic Information Systems and Science,* 2nd ed. New York: John Wiley & Sons.

Lotov, A.V., Bushenkov, V.A., Chernov, A.V., Gusev, D.V. & Kamenev, G.K., 1997. Internet, GIS and interactive decision maps. *Journal of Geographic Information and Decision Analysis,* 1 (2), 118-143.

Maguire, D.J., 1991. An Overview and definition of GIS. *In* D.J. Maguire, M.F. Goodchild & D.W. Rhind (eds.) *Geographical Information Systems, Volume 1 : Principles.* Longman, 9-20.

Malczewski, J., 1999. *GIS and Multicriteria Decision Analysis* New York: John Wiley & Sons.

Malczewski, J., 2006. Integrating multicriteria analysis and geographic information systems: the ordered weighted averaging (OWA) approach. *International Journal of Environmental Technology and Management,* 6 (1/2), 7-19.

Malczewski, J., Chapman, T., Flegel, C., Walters, D., Shrubsole, D. & Healy, M.A., 2003. GIS - multicriteria evaluation with ordered weighted averaging (OWA): case study of developing watershed management strategies. *Environment and Planning,* 35 (10), 1769 - 1784.

Malyankar, R., 1999. Creating a Navigation Ontology. *AAAI-99: Workshop on Ontology Management.* Orlando, FL: AAAI Press, 48-53.

Manson, S.M., 2000. Agent-based dynamic spatial simulation of land-use/cover change in the Yucatan peninsula, Mexico. *Proceedings of the Fourth International Conference on Integrating GIS and Enviromental Modeling (GIS/EM4)* Banff, Canada.

Marcussen, C.H., 2000. Mobile Phones, WAP and the Internet. 2000 ed.: Centre for Regional and Tourism Research.

Martin, N.J., St.-Onge, B. & Waaub, J.P., 1999. An integrated decision aid system for the development of Saint Charles River alluvial plain, Quebec, Canada. *International Journal of Environment and Pollution,* 12 (2/3), 264-279.

Massam, B.H., 1980. *Spatial Search* Oxford: Pergamon Press.

Mateos, C., Zunino, A. & Campo, M., 2005. Integrating Intelligent Mobile Agents with Web Services. *International Journal of Web Services Research,* 2 (2), 85-103.

Meng, L., 2001. The immediate usability of self-explaining interface for mobile users. *Journal of Geographical Sciences,* 11 (2001), 9-11.

Miller, H.J., 2003. What about people in geographic information science?*. *Computers, Environment and Urban Systems,* 27 (447-453.

Morris, A. & Jankowski, P., 2000. Combining fuzzy sets and databases in multiple criteria spatial decision making. *Flexible Query-Answering Systems.* 103-116.

Mountain, D. & Raper, J., 2001. Positioning techniques for location-based services (LBS): characteristics and limitations of proposed solutions. *Aslib Proceedings: new information perspectives,* 53 (404-412.

Müller, C., Almer, A., Stelzl, H., Bortenschlager, M. & Lassnig, M., 2008. Geodatenvisualisierung anf mobilen Geräten. *In* J. Strobl, T. Blaschke & G. Griesebner (eds.) *AGIT 2008.* Salzburg: Wichmann.

Muntermann, J., 2007. Event-driven Mobile Financial Information Services – Design of an Intraday Decision Support System Dissertation. Universität Frankfurt a. M.

Myles, G., Friday, A. & Davies, N., 2003. Preserving Privacy in Environments with Location-Based Applications. *IEEE Pervasive Computing,* 2 (1), 56-64.

Ogc, 2003. OGC Reference Model. *In* G. Percivall (ed.) *Open Geospatial Consortium Inc.*

Ogc, 2007. OpenGIS Geography Markup Language (GML) Encoding Standard. *In* C. Portele (ed.) *Open Geospatial Consortium Inc.*

Peterson, K., 1998. Development of Spatial Decision Support Systems for Residential Real Estate. *Journal of Housing Research,* 9 (1), 135-156.

Power, D.J., 2003. Defining Decision Support Constructs. *The seventh International Conference on Decision Support Systems (ISDSS 03).* Ustron, Poland.

Power, D.J., 2004. Decision Support Systems: From the Past to the Future. *Americas Conference on Information Systems.* New York, NY, 2025-2031.

Power, D.J., 2007. *A Brief History of Decision Support Systems* [online]. DSSResources.COM. Available from: http://dssresources.com/history/dsshistory.html [Accessed Access Date 2008].

Pühretmair, F., Rumetshofer, H. & Schaumlechner, E., 2002. Extended Decision Making in Tourism Information Systems. *In*: K. Bauknecht, A.M. Tjoa & G. Quirchmayr, eds. *E-Commerce and Web Technologies Third International Conference, EC-Web*, Aix-en-Provence, France: Springer, 414.

Qiao, Y., Yan, S., Shen, S., Xiao, C. & Zhao, X., 1999. Building the Spatial Decision Support Model Base *Towards Digital Earth — Proceedings of the International Symposium on Digital Earth*, Beijing: Science Press.

Raubal, M., 2006. Location-Based Decision Services - eine neuartige Form mobiler räumlicher Entscheidungsunterstützung. *VGI - Vermessung & Geoinformation,* 94 (1+2), 30-37.

Raubal, M. & Rinner, C., 2004. Multi-Criteria Decision Analysis for Location Based Services. *In*: S.A. Brandt, ed. *12th International Conference on Geoinformatics*, Gävle, Sweden, 47-53.

Reason, J., 1990. *Human Error* Cambridge Cambridge University Press.

Reichenbacher, T., 2003. Mobile Cartography – Adaptive Visualisation of Geographic Information on Mobile Devices. Technischen Universität München.

Rigaux, P., Scholl, M. & Voisard, A., 2001. *Spatial Databases: With Application to GIS* San Francisco, CA: Morgan Kaufmann.

Rinner, C., 2003. Web-based Spatial Decision Support: Status and Research Directions. *Journal of Geographic Information and Decision Analysis,* 7 (1), 14-31.

Rinner, C., 2005. Computer Support for Discussions in Spatial Planning. *In* M. Campagna (ed.) *GIS for Sustainable Development.* Taylor & Francis, 167-180.

Rinner, C., 2007. Multi-Criteria Evaluation in Support of Emergency Response Decision-Making *Joint CIG/ISPRS Conference on Geomatics for Disaster and Risk Management*, Toronto.

Rinner, C., 2008. Mobile Maps and More – Extending Location-Based Services with Multi-Criteria Decision Analysis. *In* L. Meng, A. Zipf & S. Winter (eds.) *Map-Based Mobile Services, Design, Interaction and Usability* Berlin: Springer, pp. 335-349.

Rinner, C. & Malczewski, J., 2002. Web-Enabled Spatial Decision Analysis Using Ordered Weighted Averaging (OWA). *Journal of Geographical Systems,* 4 (4), 385-403.

Rinner, C. & Raubal, M., 2005. Personalized multi-criteria decision strategies in location-based decision support. *Journal of Geographic Information Sciences,* 11 (1), 61-68.

Rinner, C., Raubal, M. & Spigel, B., 2005. User Interface Design for Location-Based Decision Services *Proceedings of Geoinformatics Conference*, Toronto, Canada.

Romero, J.R. & Vallecillo, A., 2005. Modeling the ODP Computational Viewpoint with UML 2.0: The Templeman Library Example. *The Enterprise Computing Conference.* Enschede, The Netherlands.

Rouse, L.J., Bergeron, S.J. & Harris, T.M., 2007. Participating in the Geospatial Web: Collaborative Mapping, Social Networks and Participatory GIS. *In* A. Scharl & K. Tochtermann (eds.) *The Geospatial Web: How Geobrowsers, Social Software and the Web 2.0 are Shaping the Network Society.* London: Springer.

Saaty, T.L., 1980. *The analytic hierachy process* New York: McGraw-Hill.

Saltf, C.A. & Dunsmore, M.C., 2000. Development of a spatial decision support system for post-emergency management of radioactively contaminated land *Journal of Environmental Management,* 58 (3), 169-178

Schiller, J.H. & Voisard, A., 2004. *Location-Based Services* San Francisco: Elsevier Science & Technology Books.

Sengupta, R.R. & Bennett, D.R., 2003. Agent-based modelling environment for spatial decision support. *International Journal of Geographical Information Science,* 17 (2), 157-180.

Shekhar, S. & Chawla, S., 2003. *Spatial Databases, A Tour* Upper Saddle River, New Jersey: Prentice Hall.

Shneiderman, B., 1998. *Designing the User Interface, Strategies for Effective Human-Computer Interaction,* 3 ed. Reading, Massachusetts: Pearson Education.

Silver, M., 1991. *Systems that support decision makers: description and analysis* New York: Wiley

Simon, H.A., 1960. *The new science of management decision* New York: Harper & Row.

Simon, R., 2008. Mobilizing the Geospatial Web: A Framework an Conceptual Model for Spatially Aware Mobile Web Applications. Technische Universität Wien.

Sprague, R., 1980. A Framework for the development of Decision Support Systems. *MIS Quarterly,* 4 (4), 1-26.

Sugumaran, R., Meyer, J., Barnett, Y.Z., Prato, T. & Fulcher, C., 2003. A Web-Enabled Spatial Data Visualization and Decision Support System. *7th world multi conference on Systemics, Cybernetics and Informatics.* Florida, USA, 390-394.

Sugumaran, V. & Sugumaran, R., 2003. Spatial Decision Support Systems Using Intelligent Agents and GIS Web Services. *Americas Conference on Information Systems.* Tampa, Florida, 2481-2486.

Sugumaran, V. & Sugumaran, R., 2005. Web-based Spatial Decision Support Systems (WebSDSS): Evolution, Architecture, and Challenges. *In*: K. Corral & D. Schuff, eds. *Third Annual SIGDSS Pre-ICIS Workshop: Designing Complex Decision Support: Discovery and Presentation of Information and Knowledge*, Las Vegas, Nevada.

Tao, V., Liang, S., Croitoru, A., Haider, Z.M. & Wang, C., 2004. GeoSWIFT: Open Geospatial Sensing Services for Sensor Web. *In* A. Stefanidis & S. Nittel (eds.) *GeoSensor Networks.* Boca Raton, Florida: CRC Press.

Turner, A.J., 2006. *Introduction to Neogeography* Cambridge, MA: O'Reilly.

Uhlirz, M., 2007. A Market and User View on LBS. *In* G. Gartner, W. Cartwright & M.P. Peterson (eds.) *Location Based Services and Tele-Cartography.* Heidelberg, Berlin: Sringer, 47-58.

Uran, O. & Janssen, R., 2003. Why are spatial decision support systems not used? Some experiences from the Netherlands. *Computers, Environment and Urban Systems,* 27 (5), 511-526

Vineke, P., 1992. *Multicriteria Decision-aid* West Sussex: John Wiley & Sons.

Virrantaus, K., Tirri, H., Veijalainen, J., Markkula, J., Katanosov, A., Garmash, A. & Terziyan, V., 2001. Developing GIS-Supported Location-Based Services *2nd International Conference on Web Information Systems Engineering (WISE 2)*, Kyoto, Japan: IEEE Computer Society, 66-75.

Wang, L. & Cheng, Q., 2007. Design and implementation of a web-based spatial decision support system for flood forecasting and flood risk mapping *IGARSS 2007, Geoscience and Remote Sensing Symposium*, Barcelona, Spain: IEEE International, 4588 - 4591.

Wang, L. & Cheng, Q.M., 2006. Web-Based Collaborative Decision Support Services: Concept, Challenges and Application. *ISPRS Technical Commission II Symposium.* Vienna: International Archives of Photogrammetry, Remote Sensing, and Spatial Information Sciences Vol. XXXVI - Part 2, 76-84.

Want, R., Schilit, B.N., Adams, N.I., Gold, R., Peterson, K., Goldberg, D., Ellis, J.R. & Weiser, M., 1995. An Overview about ParcTab Ubiquitous Computing Experiment. *IEEE Personal Communications,* 2 (6), 28-43.

Worboys, M.F. & Duckham, M., 2004. *GIS A Computing Perspective,* 2nd ed. Boca Raton: Taylor & Francis.

Wu, F., 1998. SimLand: a prototype to simulate land conversion through the integrated GIS and CA with AHP-derived transition rules. *International Journal of Geographical Information Science,* 12 (1), 63-82.

Yan, S., Tian, Q., Xiao, Q.C., Zhao, W.J. & Zhou, Y., 1999. Spatial Decision Support System and its Software tools. *In* B. Li (ed.) *Geoinformatics'99 Conference.* California, Berkeley: Hilgard Hall, 1-6.

Zipf, A. & Strobl, J., 2002. *Geoinformation mobil* Heidelberg: Wichmann Verlag.